

(3/86)
COMMODORE WELT
DM14,80
ÖS124
SFR14,80
C 16
116
P/4
SPECIAL
Das grosse Sonder-Heft für C16/116/Plus 4!
Service
Tips &
Tricks
Listings
MISSILE
AVOIDED
CANNON
COLLISION
WARNING
Im Test:
Software für
C16+Plus 4

# INHALT 16 er SPECIAL





Die Floppy 1551, speziell für die „kleinen" Commodores gedacht, kostet weniger als 400 DM!


Der Userport des Plus 4 kann noch zu vielen anderen Dingen genutzt werden — wir berichten




Bücher für die 16er-Baureihe gibt es en masse — wir haben für Sie gelesen

## TESTS

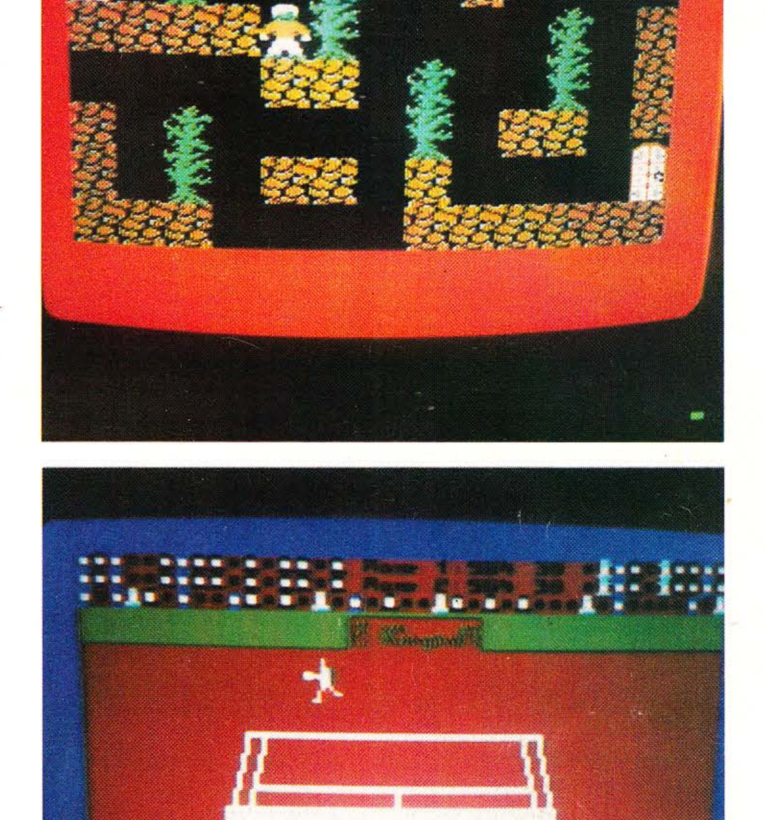


## SERVICE, TIPS & TRICKS

# INHALT 16er SPECIAL

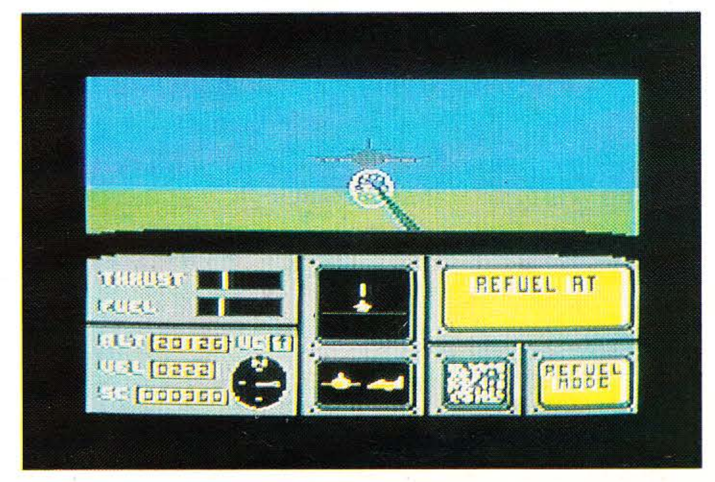
## LISTINGS



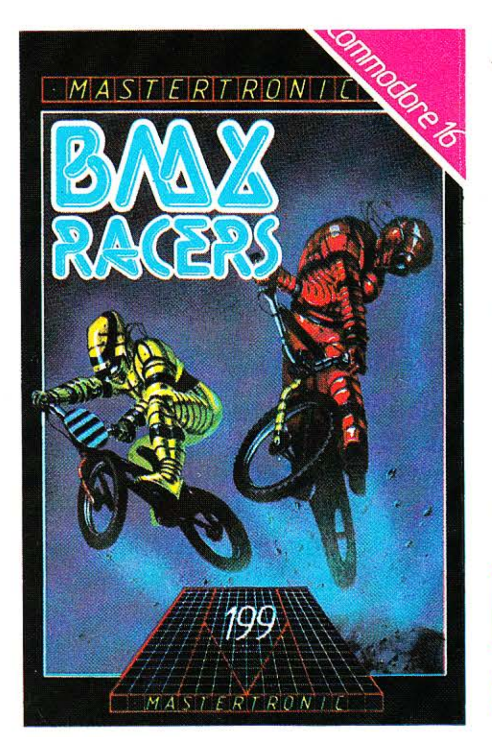


**Bongo — ein unterhaltsames Spiel — wir haben es getestet**

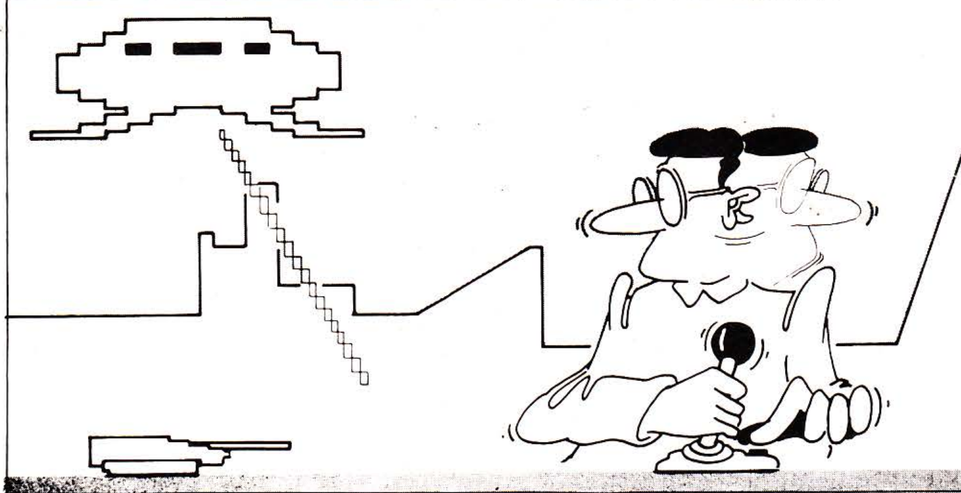
**Über sieben Seiten allein für Software-Tests — in dieser Spezialausgabe**




**Der Überhammer: ACE läuft gleichzeitig auf dem Plus 4 und auf dem C 64. Von uns geprüft.**




BMX Racers



# NEUE SPIELE GESICHTET!

**Mit wachsender Verbreitung der 3.5er-Rechner von Commodore wächst auch das Angebot an Spielesoftware. Zwar ist dieses immer**

## THE EXPLOITS OF FINGERS MALONE

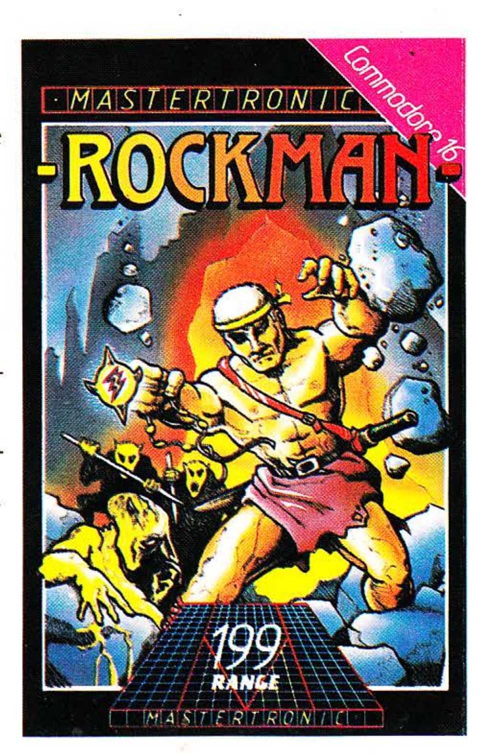
Instruktionen: Englisch.
Die Spielfigur muß in einem Klettergerüst mit Aufzügen verschiedene Schlüssel einsammeln, um mit diesen einen Safe knacken zu können. Hierbei sind allerdings zwei Minimonster sehr hinderlich, welche den Safeknacker bei Berührung verspeisen. Ist ein Safe geöffnet, gelangt die Figur in die nächste Runde von 15. Das Spiel läuft sehr schnell ab, so daß man anfangs kaum über die erste Spielstufe hinauskommt. Die Grafik muß als mittelmäßig bezeichnet werden, allerdings bleibt der Spielspaß davon unverschont.
Ein weiteres Spiel auf dem "Baugerüst".
**Bedingt empfehlenswert**

## ROCKMAN + THE RETURN OF ROCKMAN

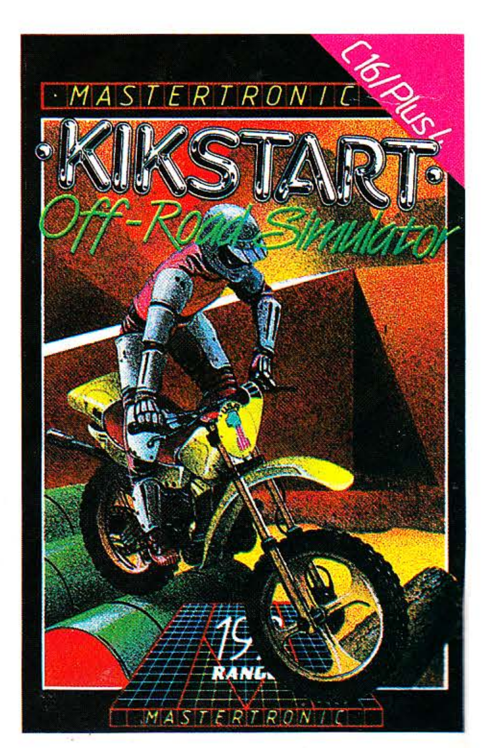
Instruktionen: Deutsch.
Die Figur muß, ähnlich dem bekannten Automatenspiel, in unterirdischen Stollen Diamanten aufsammeln. Gräbt er dabei zu nahe an den verstreuten Felsbrocken, fallen diese hinunter und bei zu langsamer Reaktion ihm auf den Kopf.
Dieses Spiel kommt dem bekannten Original schon recht nahe und ist auch über längere Dauer interessant. Die Fortsetzung des Spieles, "The Return of Rockman", ähnelt der ersten Version sehr stark, lediglich der Bildschirmaufbau variiert etwas. Mit der Grafik hätte man sich im Hause Mastertronic zwar etwas mehr Mühe geben können, das beeinträchtigt den Spaß jedoch kaum.
**Empfehlenswert**

## VEGAS JACKPOT

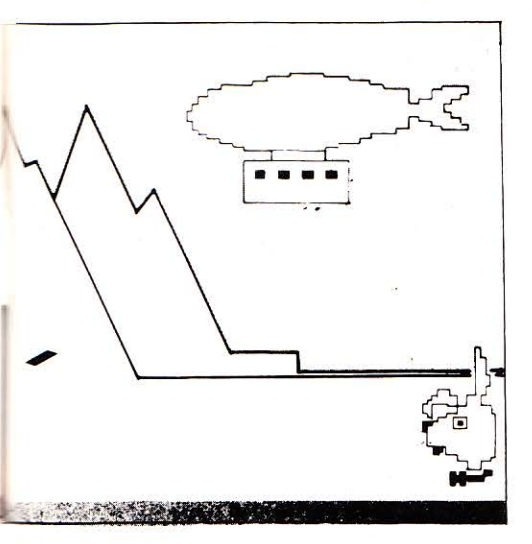
Instruktionen: Deutsch.
Dies ist eine absolut gelungene Nachempfindung der bekannten einarmigen Banditen aus Las Vegas. Auf vier verschiedenen Scheiben rotieren Früchte mit unterschiedlicher Bedeutung und damit auch




Rockman




Kikstart



**noch meilenweit vom legendären C 64 entfernt, es ist jedoch offensichtlich, daß auch für C 16/116/ Plus 4-Eigner rosige Spielzeiten angebrochen sind. Wir nahmen einige Spiele der beiden großen Anbieter auf dem deutschen Markt, Kingsoft und Mastertronic, unter die Lupe.**

unterschiedlicher Gewinnmöglichkeit. Zuerst zählen natürlich mehrere gleiche Früchte in Reihenfolge. Hierdurch gelangt man auf die "Risikoleiste", auf welcher ständig der mögliche Gewinn hin- und herblinkt. Geht man das Risiko ein, kann der Gewinn bis zu verzehnfacht werden, allerdings ist es natürlich auch möglich, daß der Spieler leer ausgeht. Wer das Risiko scheut, kann auch den kleinen, aber sicheren Betrag übernehmen. Stehen neben den Früchten Zahlen und beträgt die Gesamtsumme mehr als sechs, gelangt man in die obere Risikoleiste, in der man sich für die Gewinnausspielung ähnlich wie vorher qualifizieren kann. Erreicht man hierbei durch Übernehmen oder eben durch Risikobereitschaft eine Nummer, darf in der unteren Risikoleiste der Gewinn erhöht oder übernommen werden. Je nach der oben erreichten Nummer gelangt man unten in eine entsprechend vorgerückte Position auf der Gewinnskala, in der dann wie anfangs beschrieben gewonnen werden kann. Der Anfangsbesitz liegt bei 100 Pfund, erreicht der Spieler 250, so hat er den Jackpot gewonnen. Mit der überaus gelungenen Grafik und der wirklichkeitsgetreuen Funktionsweise stellt dieses Spiel einen echten Ersatz für diejenigen dar, die schon immer gerne einen "Einarmigen" im Wohnzimmer haben wollten.

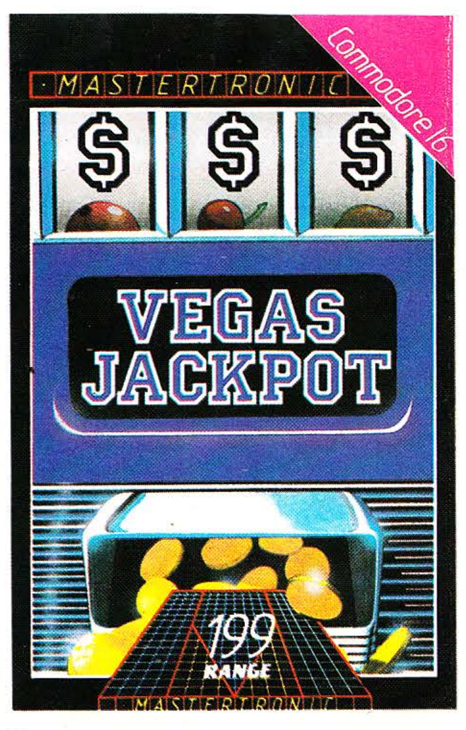
**Sehr empfehlenswert**

PS: Leider ist es uns bisher noch nicht gelungen, den Computer zur Auszahlung von Bargeld zu bewegen.

## SPECTIPEDE

Instruktionen: Deutsch.
Dieses Spiel (der Name verrät es bereits) ist dem bekannten Klassiker Centipede nachempfunden. Es müssen durch verschiedene Schwierigkeitsstufen hindurch alle möglichen Krabbeltierchen, wie Spinnen usw., beseitigt werden. Erschwert wird dies durch Hindernisse, welche abgeschossen werden müssen und durch den "Plasmabolzen" ständig wieder aufgebaut werden. Dieser ist übrigens resistent gegen sämtliche Schüsse, außerdem reflektiert er diese gemeinerweise. Durch die hervorragende Grafik und sehr hohe Spielgeschwindigkeit ist dies ein gelungener "Nachbau" von Centipede auf dem C 16.

**Sehr empfehlenswert**

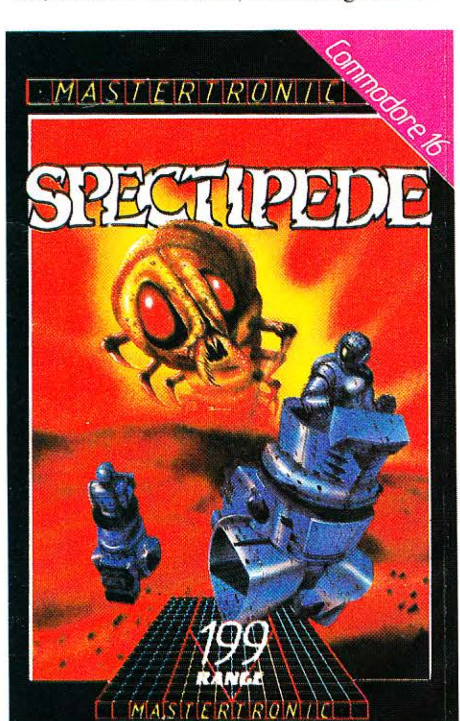



Vegas Jackpot

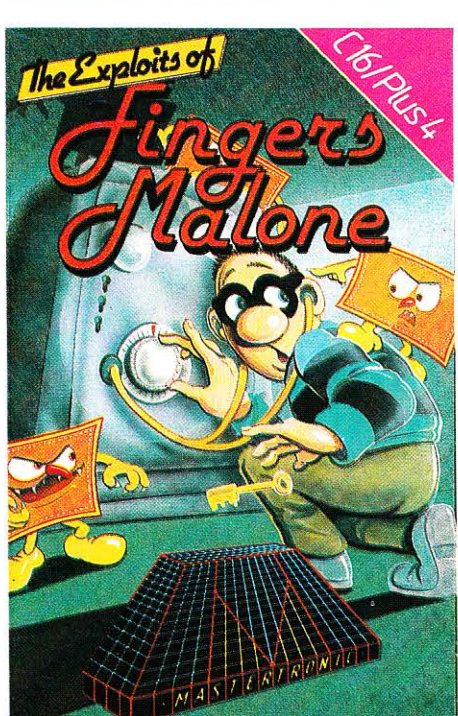
## BMX-RACERS

Instruktionen: Deutsch.
Auf einem Kurs, welcher von oben dargestellt wird, muß das BMX-Rad gesteuert werden. Hierbei müssen Energiepunkte eingesammelt und Markierungsfähnchen überfahren werden. Durch ein Energie- (=Zeit)limit ist es relativ schwierig, ins Ziel zu gelangen.
Hier hindern noch zusätzlich die zu hohe Geschwindigkeit und die mangelnde Lenkpräzision. Die absolut mißlungene Grafik (das angebliche BMX-Rad ist ein hingeklatschter Fleck) macht das Spiel auch nicht schöner. Leider ist der Schwierigkeitsgrad nicht einstellbar, so daß der Spaß bald vergangen ist.

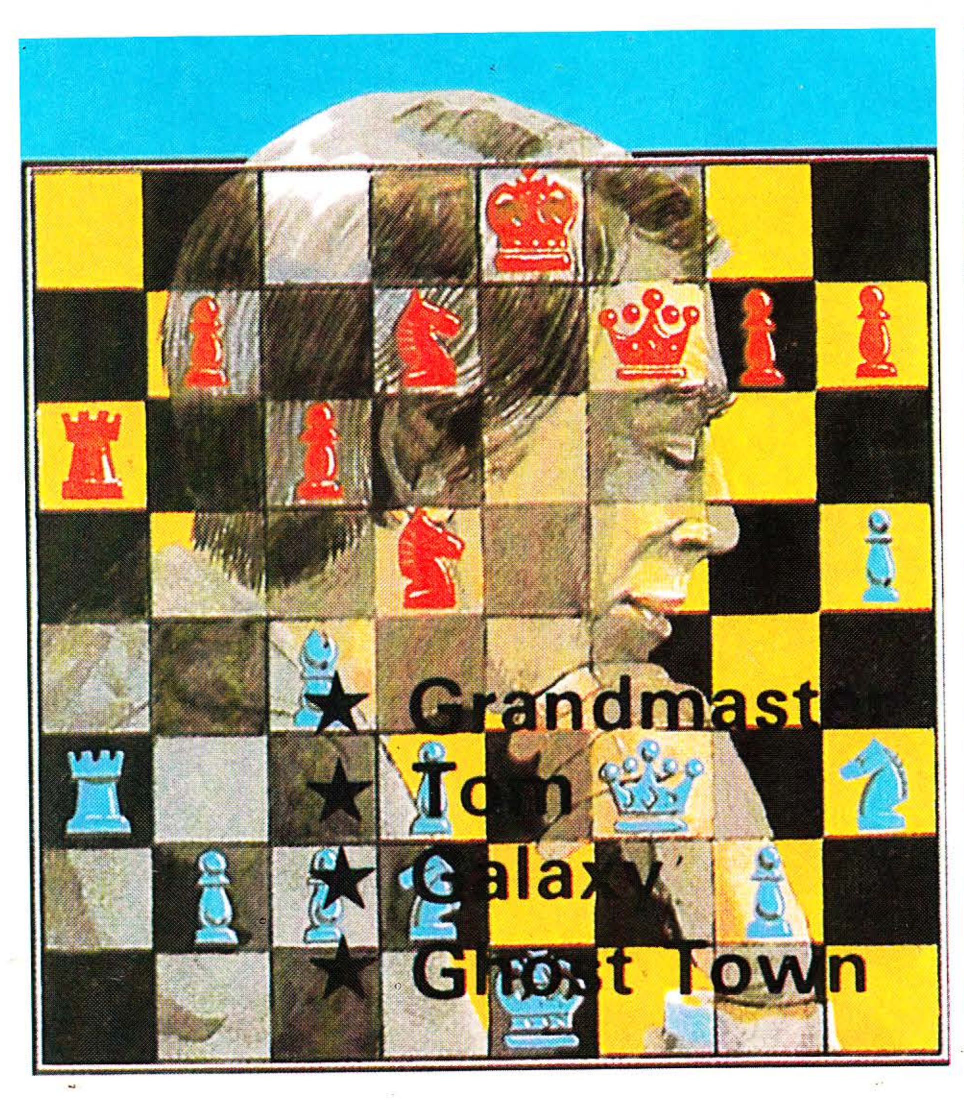
**Nicht empfehlenswert**




Spectipede




Finger's Malone





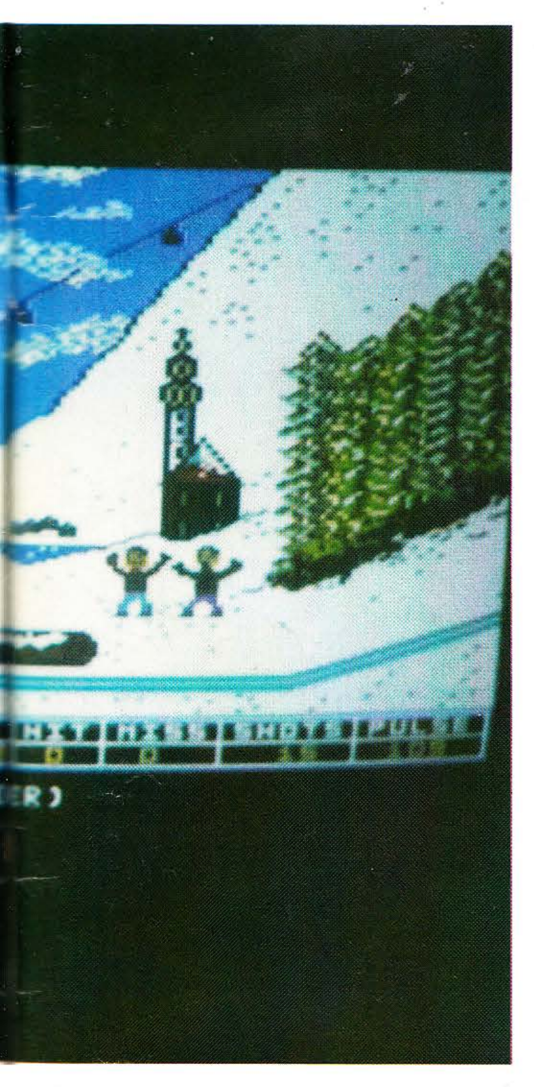


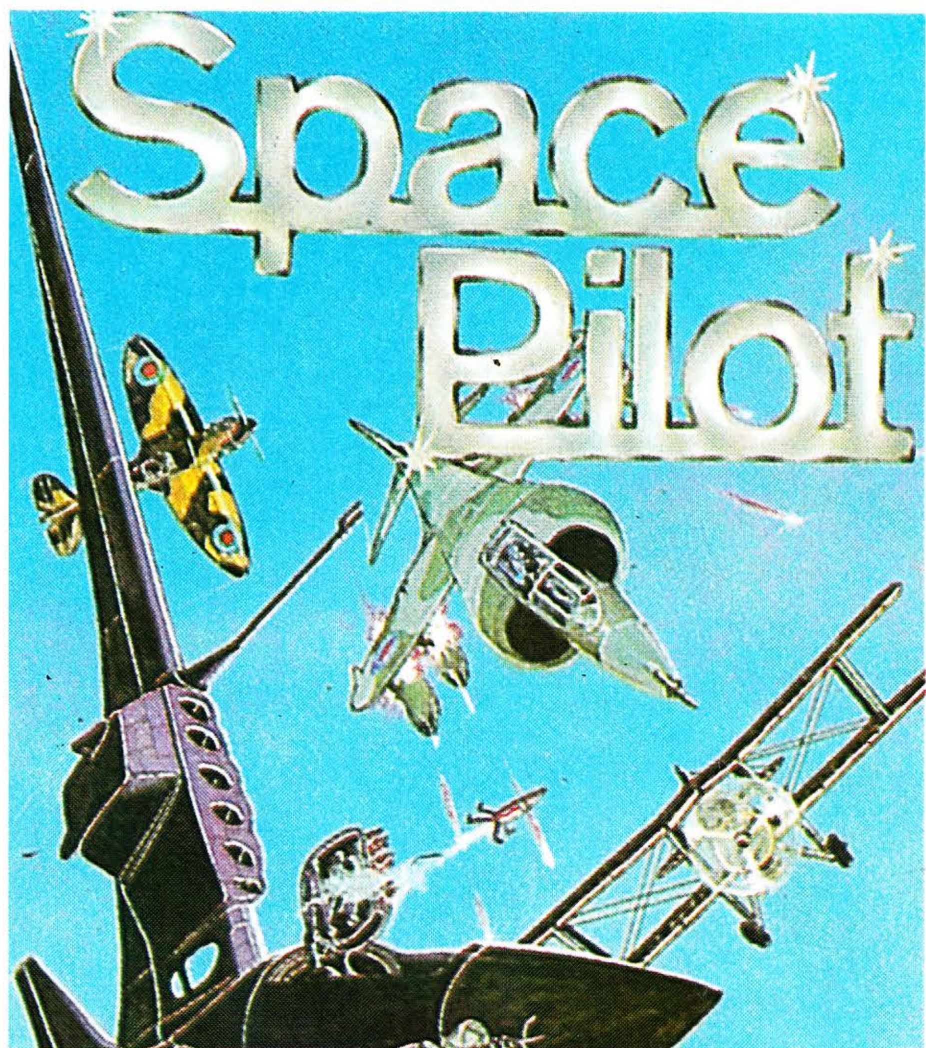
# KINGSOFT: WER SAGT DENN, DASS NUR DIE ENGLÄNDER GUTE PROGRAMME HABEN?

**Daß in England hervorragende Programmautoren beheimatet sind, ist inzwischen wohl hinlänglich bekannt. Das deutsche Softwarehaus Kingsoft allerdings beweist, daß sich auch die inländischen Programmierer nicht zu verstecken brauchen. Kingsoft bietet eine ganze Reihe von Spielen auf den 3.5er-Modellen an. Einige hiervon im Test.**

## SPACE PILOT

Ein weiteres Abenteuer im Weltraum. Von oben wird das fliegende Schiff dargestellt, dessen Aufgabe es ist, verstreut umherstreifende Gegner aufzuspüren und zu vernichten. Vom Bildschirmaufbau und der Grafik her absolut hervorragend, muß dieses Spiel sich jedoch auch Kritik gefallen lassen: Die Steuerung ist äußerst gewöhnungsbedürftig und in puncto Geschwin-








digkeit stellt Galaxy mit Sicherheit keine Rekorde auf.
Trotzdem ein interessantes Spiel.
**Empfehlenswert**

## GRANDMASTER

Schachsimulation. Ein hervorragendes Schachprogramm für die C16/C116/Plus 4-Reihe hat Kingsoft auch zu bieten.
Natürlich sind alle Möglichkeiten anderer Schachsimulationen ebenfalls enthalten (z.B. Tip oder Erstellen einer eigenen Partie). Für den Schachliebhaber und C 16–Besitzer ein Muß!
**Sehr empfehlenswert**

## BONGO

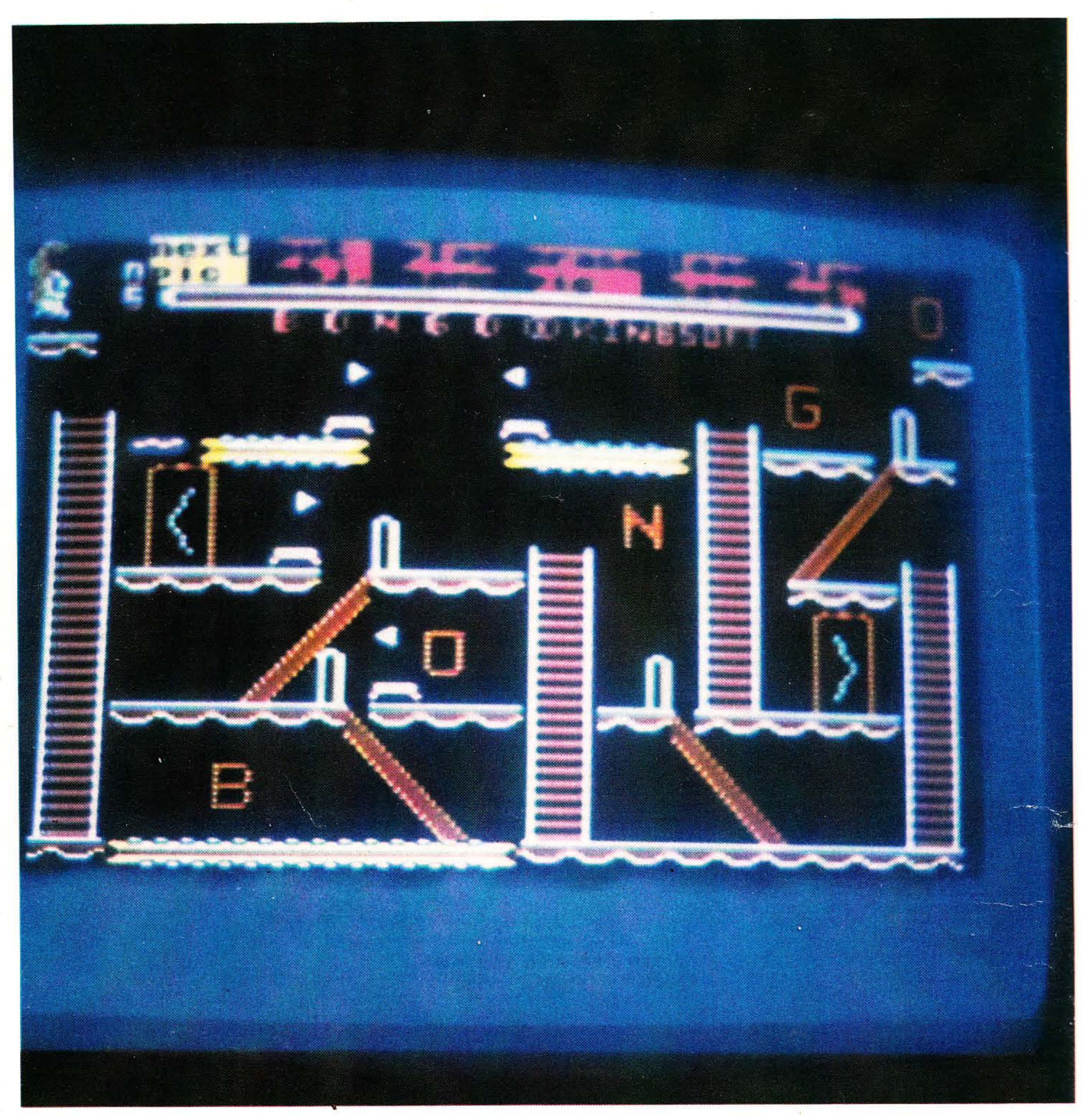
Auf einem Klettergerüst mit Rutschen, Leitern und Aufzügen muß die Spielfigur versuchen, herumfliegende Buchstaben aufzusammeln, um den Namen "Bongo" zu buchstabieren. Hierbei kommt dem Helden aber ein langnasiger Schurke ins Gehege, welcher nichts Besseres zu tun hat, als diesen vom Gerüst zu werfen.
Dieses Spiel stellt in jeder Hinsicht eine Bereicherung der eigenen Sammlung dar. Angefangen vom absolut gelungenen Sound, als Anfangsmelodie wird das bekannte "Axel F."-Stück gespielt, über die hervorragende Grafik bis hin zur Spielidee und Durchführung verdient Bongo uneingeschränktes Lob.
**Sehr empfehlenswert**

## GHOST TOWN

In einem unterirdischen Labyrinth muß durch viele Räume hindurch versucht werden, ein geheimes Lösungswort zu finden, um den bösen Zauberer zu vernichten. Die Grafik ist recht gelungen, jedoch kommt es durch die großräumige Steuerung öfters zu Kollisionen mit Dornenbüschen und dergleichen, was zum Spielende führt. Daher ist bei Ghost Town einige Übung erforderlich, um dem Ziel wenigstens auf Reichweite nahe zu kommen. Der Spaß bleibt mit Sicherheit über längere Zeit ungetrübt.
**Empfehlenswert**

## GALAXY

Die inzwischen wohl ausreichend ausgelutschte Idee vom einsamen Raumschiff, welches sich unbarmherzigen Angreifern stellen muß, ist hier zu einem doch recht gelungenen Spiel verarbeitet worden. In bekannter Weise treibt das eigene Schiff am unteren Bildschirmrand hin und her und versucht, die Angreifer, welche in immer neuen Formationen und Variationen angreifen, zu vernichten. Da tauchen ebenso furchterregende wie unbarmherzige Elmorcs auf, auch mit Forculs (hoffentlich weiß wenigstens der Autor, was die ganzen Namen bedeuten) darf sich der einsame Streiter beschäftigen. Ein recht gelungenes Spiel, welches trotz der bekannten Idee durch Abwechslungsreichtum bestícht. Uns jedenfalls ist es nicht gelungen, irgendeine Angreiferformation wiederholt zu sichten.
**Empfehlenswert**




## LEGIONÄR

Hier hat eindeutig das 64er-Spiel Rambo Pate gestanden. Ähnlich wie in diesem geht es auch bei Legionär darum, möglichst viele Gegner per Maschinengewehrsalven ins Jenseits zu befördern. War Rambo schon als Film und erst recht als Telespiel umstritten, muß sich nun auch Legionär solche Kritik gefallen lassen. Natürlich bleibt jedem selbst überlassen, ob er am Monitor seine Mordgelüste austoben will, ein Spiel für das Kinderzimmer ist Legionär jedoch wohl kaum. Hier hilft auch nicht, daß die Grafik der Landschaft sehr gut ausgefallen ist.
Letztendlich bleibt es wohl beim Käufer selbst, ob er dieses Spiel erwerben möchte.
**Kein Urteil**

## WINTER OLYMPIADE

Auf dem inzwischen sehr gefragten Sport-Spiele-Sektor hat Kingsoft neben einem Tennisspiel auch die besagte Winterolympiade zu bieten. Eigentlich ist es aber ein Dreikampf, denn diese Winterolympiade besteht aus den drei (schönsten) Disziplinen Biathlon, Skisprung und Bobfahrt. Im Bildschirmaufbau absolut identisch mit den berühmten Winter Games des 64ers, kann die Grafik natürlich nicht ganz an das Original heranreichen, ist jedoch trotzdem recht gelungen.
**Sehr empfehlenswert**

Die Kingsoft-Spiele sind auf Kassette oder Diskette erhältlich. Preise erfahren Sie auf Anfrage.

# TIPS & TRICKS: KEY-EIN SCHLÜSSEL-WORT!

## FUNKTIONSTASTENBELEGUNG ABSPEICHERN

Hat man sich die Tastenbelegung für eigene Anwendungen entsprechend abgeändert und will diese auch später wieder verwenden, so kann man sie sich abspeichern. Dazu gibt es zwei Möglichkeiten. Zum einen kann man den Speicherbereich 1375–1510 ($055f–$ 05e6) mit dem Monitor abspeichern. Dies macht man mit folgenden Befehlen:

```
monitor                  (RETURN)
s"key",8,055f,05e7       (RETURN)
x                        (RETURN)
```

Bei Verwendung der Datasette muß man die 8 durch 1 ersetzen.
Nun kann man mit:

```
load"key",8,1            (RETURN)
new                      (RETURN)
```

oder mit Monitor mit:

```
l"key",8                 (RETURN)
```

diese Tastenbelegung jederzeit wieder einladen.
Die zweite Möglichkeit besteht darin, die KEY-Definition in ein Basic-Programm zu schreiben. Dabei kann man sich durch einen kleinen Trick viel Tipparbeit ersparen.
Geben Sie ein:

```
key                      (RETURN)
```

Jetzt stehen ja die ganzen Belegungsbefehle auf dem Schirm, es fehlen nur noch die Zeilennummern. Dazu geht man mit dem Cursor auf das k von key, überschreibt es mit 'auto10' und drückt die RETURN-Taste. Mit ESC&A schalten Sie den Einfügemodus ein und schreiben 10. Dann muß man nur noch 7mal RETURN drücken und schon hat man ein kleines Basic-Programm zur Tastenbelegung, das man wie gewohnt abspeichern kann.

## ALLES GUTE GEHT NOCH BESSER

Die Originalbelegung der Funktionstasten ist gut durchdacht.
Man kann sich aber durch leichte Änderungen beim Programmieren viel Zeit sparen. Will man den DIRECTORY-, den LIST- oder den RUN-Befehl verwenden, so muß man sich ja immer zuerst eine freie Zeile schaffen, bevor man die f8-Taste drückt, da man sonst einen SYNTAX ERROR erntet.
Schreibt man aber ein 'CHR$ (147)+' (Bildschirm löschen) vor den Befehl, kann man die Taste auch drücken, wenn sich der Cursor mitten in einer beschriebenen Zeile befindet.
Übrigens wird das 'CHR$(147)+' automatisch in ein reverses Herzchen umgewandelt, um Speicherplatz zu sparen. Man macht die Änderung also am leichtesten so: Zuerst mit dem Cursor auf das 'L' von 'LIST' fahren, dann 1mal SHIFT/INST, 1mal SHIFT/CLR und RETURN drücken. Ein weiterer Trick ist beim DLOAD-Befehl auf f2 angebracht. Meist listet man sich doch zuerst das Directory auf, um zu sehen, wie das File, welches man laden will, genau heißt. Nun kann man mit dem Cursor an den Anfang der Zeile fahren, in der der Filename steht und f2 drücken. Damit man nun aber eine gültige Eingabe erhält, muß man hinter den Filenamen noch einen ':' setzen. Dazu muß zuerst mit ESC&0 der Anführungszeichenmodus ausgeschaltet werden, damit man mit dem Cursor nach rechts fahren kann.
Dies alles kann man natürlich auch gleich in die Tastenbelegung aufnehmen. Wie, zeigt ein Beispiel für eine besonders sinnvolle Tastenbelegung, welche den „krönenden Abschluß" bildet.

## TASTENBELEGUNG PAR EXCELLENCE

Hier ist ein kleines Basic-Programm, das eine sehr hilfreiche Funktionstastenbelegung liefert:

```
10 rem ***key-belegung**********
15 cr$=chr$(13):cl$=chr$(147
20 fori=1to21:p$=p$+chr$(29):next
25 key 1,"gRO"+cr$
30 key 2,"dL"+p$+":"+cr$
35 key 3,cl$+"diR"+cr$
40 key 4,"sC"+cr$
45 key 5,"dS"+chr$(34)
50 key 6,cl$+"rU"+cr$
55 key 7,cl$+"lI"+cr$
60 key 8,"heL:wA198,63:sY1470"
        +cr$
```

Hat man die Tastenbelegung fertig, sollte man sie sich wie oben beschrieben mit dem Monitor abspeichern. Vorher muß jedoch noch das kleine Maschinenprogramm 'FLASH WEG' aus dem Artikel über den Monitor eingeladen werden (S. 16).
Dadurch hat man die HELP-Taste wesentlich verbessert. Drückt man jetzt nach einem Fehler im Programm die f4-Taste, so erscheint die fehlerhafte Zeile, wobei der Teil mit dem Fehler blinkt. Da dieses Blinken auf die Dauer sehr lästig ist, wartet nun das Miniprogramm mit WAIT198,63 auf einen Tastendruck, um dann mit SYS1470 das Blinken abzustellen.

»

# HÄTTEN SIE ES GEWUSST? TIPS & TRICKS

Vorneweg ein paar Bemerkungen zum Literaturangebot für diese Computertypen: In vielen Publikationen wird behauptet, es gebe so gut wie keine. Das ist falsch. Außer den neuerdings besser gewordenen Handbüchern gibt es allein bei Data Becker über sechs Bücher, die sich mit diesen Computern beschäftigen.
Im Vergleich zum VC-20 oder C64 ist das zwar nicht viel, aber auch die haben mal klein angefangen. Auch das Angebot an Software wird immer besser. Auf dem Spielesektor ragen vor allem die Firmen Kingsoft und Mastertronic hervor. Die Firma Kingsoft bietet außer den Spieleprogrammen auch Hardwareerweiterungen für die Kleinen von Commodore an. Es sind eine 16K- und eine 64K-Erweiterung, wobei letztere in den Computer eingebaut wird und damit immer präsent ist.
Außer Spielen bietet diese Firma auch noch eine Datenbank, eine Textverarbeitung (in Maschinensprache), ein Mal- und ein Musikprogramm an.

## TURBOTAPE FÜR C 16

Zu den bisherigen Programmen gesellte sich ein Turbo-Tape, das den C-16/C-116/Plus 4 ein ca. achtmal schnelleres Laden und Speichern auf Kassetten erlaubt. Auch die manchmal schwer zu erhaltenden Joystickadapter kann man hier bekommen. Eine 16K-Hardwareerweiterung gibt es auch bei der Firma Jeschke.
Schmeißt man sich mal in Unkosten und investiert in ein ausländisches Computermagazin, gehen einem beim Studium die Augen auf. Da annoncieren C16/Plus 4 Center, es werden Spiele besprochen und auch Artikel über diese Computer findet man dort. In einer Anzeige zählte ich über 100 Programme für den C-16 bzw. Plus 4. Selbst in der Spielehitparade werden die Spiele für diesen Computer berücksichtigt.
Nun ein paar Tips, die den Umgang mit dem C-16/Plus 4 erleichtern sollen.
Viele, die von einem anderen Commodore-Computer umgestiegen sind, vermissen vielleicht die Restoretaste, die ein Aussteigen aus dem Programm und damit einen Warmstart ermöglicht. Auch beim C-16/Plus 4 ist dies möglich. Drücken Sie zuerst die RUN/STOP-Taste, halten diese gedrückt und betätigen dann die RESET-Taste. Damit landen Sie im eingebautem Maschinensprachemonitor, den Sie dann mit der Befehlsfolge x >RETURN< wieder verlassen können. Damit ist das Programm wieder da.
In einem Sonderheft einer renommierten Computerzeitung steht es leider umgekehrt.
Der eingebaute Maschinensprachemonitor ist zwar sehr nützlich, aber meiner hat einige Mucken. Mir passiert es leider öfters, daß das Programm aus unerklärlichen Gründen in den Monitor springt. Dieses ist zwar nicht weiter schlimm, aber immer unangenehm.
Auch für die Besitzer der Diskettenstation hält der Computer gelegentlich eine „Überraschung" bereit. Vielleicht ist es Ihnen auch schon einmal passiert, daß plötzlich, während des Programmierens, ein Programm geladen wurde.
Wahrscheinlich haben Sie ganz aus Versehen die SHIFT und die RUN/STOP-Taste betätigt. Da die Computer C-16/Plus 4 ganz auf Diskette ausgerichtet sind, kann mit dieser Tastenkombination das erste Programm der Diskette geladen und gestartet werden. Beim VC-20 und C64 wird hiermit der Cassettenrecorder angesprochen, der aber nicht, wie die Floppy, von alleine losläuft. Ich jedenfalls nehme die Diskette nach dem Laden oder Saven immer heraus, denn nichts ist ärgerlicher als ein schon fast fertiges und dann überschriebenes Programm.

## FUNKTIONSTASTEN FÜR CASSETTENBETRIEB

Auch Cassettenbesitzer können die mit den Ladebefehlen für die Diskettenstation belegten Funktionstasten (F2/F5) für Cassettenoperationen wie Saven und Laden verwenden. Bevor Sie eine dieser Tasten betätigen, drücken Sie einmal die Escape-Taste und dann die entsprechende Funktionstaste. Versuchen Sie es einmal. Sie sparen sich eine Menge Tiparbeit.

*B. Welte*

## FRIEREN SIE DOCH EIN!

Für die Besitzer von VC-20 und C-64 sieht es trübe aus. Aber die Fans der 16er-Serie und des Plus 4 können sich freuen, denn das Basic dieser Computer hält eine Methode bereit, mit der dieser Effekt möglich ist. Mit dem Befehl CHR$ (27)"T" wird der obere Rand des Bildschirmfensters festgelegt. Wenn nun dieser Befehl hinter einem Printbefehl steht, wird der Printbefehl in den oberen Zeilen eingefroren, da das Fenster (in Neudeutsch Window) nicht bis an den oberen Bildschirmrand reicht.

## LISTEN IM PROGRAMM

Geht so etwas?
Im Basic 7,0 des PC-128 ist es zum Beispiel möglich, den Programmablauf zu unterbrechen, um einen Teil oder das ganze Programm auflisten zu lassen. Aber auch schon im 3,5 Basic der 16er-Reihe besteht diese Möglichkeit. Man setzt ein-

fach an die Stelle, an der gelistet werden soll, den Befehl >LIST<, entweder alleine oder mit entsprechenden Zeilennummern versehen, und wartet auf das, was da kommt. Wenn das Programm den Befehl erreicht, werden die entsprechenden Zeilen gelistet und das Programm fährt anschließend in seiner Arbeit fort. Um das Listen zu verfolgen, sollte man noch eine FOR-NEXT-Schleife anhängen, damit man das Listing besser beobachten kann. Sie können sich die gelisteten Zeilen sogar auf dem Drucker ausgeben lassen.
Solch eine Zeile kann z.B. so aussehen:
XXXX OPEN4,4:CMD4 :LIST
10-20:PRINT #4:CLOSE4
Nach dem Ausdrucken fährt das Programm mit seiner Arbeit weiter fort.



## MONITOR AUSGEDRUCKT

Vielleicht haben Sie schon einmal gedacht: Könnte ich doch nur den Inhalt des Maschinensprachemonitors als Dokumentation ausdrucken! Leider hat der Monitor keinen entsprechenden Befehl, um diese Arbeit zu erledigen. Aber es geht doch. Geben Sie im Direktmodus folgende Zeilen ein. Open4,4: CMD4
Anschließend gehen Sie in den Monitor. Ihr Drucker wird sich jetzt schon mit >Monitor und Ready< melden. Jetzt geben Sie einfach den auszudruckenden Bereich an, z.B. D10001 FFF oder M 1000 1FFF, und schon wird der Inhalt dieser Speicherstellen auf dem Drucker ausgegeben. Zum Schluß verlassen Sie den Monitor wieder mit X und schließen den Druckerkanal mit >Print#4:Close4<

## DRUCKER-DIRECTORY OHNE PROGRAMM-VERLUST

Um sich eine Directory ausdrucken zu lassen, mußte man sie bisher in den RAM-Speicher laden. Dadurch wurde ein im Speicher stehendes Programm überschrieben. Das muß nicht sein. Leiten Sie die Directory einfach gleich auf den Drucker. Sie bekommen einen Ausdruck und das Programm ist auch noch heil. Um dieses zu erreichen, geben Sie einfach folgende Zeile ein:
>OPEN4,4:CMD4: DIRECTORY<
>RETURN<
Zum Schluß müssen Sie nur noch den Druckerkanal mit >PRINT #4: CLOSE$< >RETURN< schließen. Versuchen Sie es einmal.

## SCHLEIFENBEHANDLUNG IM BASIC 3.5

Was soll ich mit Schleifen, werden Sie sich fragen. Wofür brauche ich Schleifen beim Programmieren? Einige Fragen, die immer wieder auftauchen. Obwohl viele nicht wissen, was eine Schleife beim Programmieren ist, sind fast alle schon darüber gestolpert. Fast kein Programm kommt ohne Schleifen aus. Sei es um zu zählen, den Computer für eine bestimmte Zeit anzuhalten, um eventuell auf Eingaben zu warten, oder zum Einlesen von Daten aus Datazeilen und viele Anwendungen mehr. Vielleicht haben Sie in einem Listing schon mal den Begriff FOR...TO... NEXT gelesen. Das ist schon eine Schleife. Heute wollen wir uns aber mit der Möglichkeit der Schleifenprogrammierung, die uns das Basic 3.5 bietet, beschäftigen.
Gegenüber dem Basic der Computer VC-20 und C-64 verfügt das Basic 3.5 über die Möglichkeit des strukturierten Programmierens. Durch die neuen Schleifenbefehle des Basic 3.5 erübrigen sich die für Unübersichtlichkeit sorgenden GOTO-Sprünge. Mit den Befehlen DO, LOOP, UNTIL, WHILE und EXIT können jetzt Schleifen schnell und einfach programmiert werden. Um z.B. die oft verwendete, aber eigentlich unerwünschte, Endlosschleife zu programmieren, genügen nun nur noch zwei Befehle: DO und LOOP. Auf den ersten Blick erscheint die Schleife auch nicht kürzer als die Kombination 100 GOTO 100. Da aber die Sprungbefehle immer von 0 an hoch gezählt werden, ist die Schleife mit 100 DO:LOOP viel schneller. Mit einer Endlosschleife kann man im allgemeinen nicht viel anfangen, denn Schleifen müssen irgendwann zum Ende kommen, also endlich sein. Eine endliche Schleife mit DO LOOP sieht etwa so aus.

```
100 X20
110 DO
120 X=X+1
130 LOOP UNTIL X=10
140 PrintX
```

In dieser Schleifenversion erfolgt der Austritt aus der Schleife am Ende derselben. Denn dort steht die Bedingungsabfrage "UNTIL X=10". Diese Schleife läuft also solange, bis X den Wert 10 hat. Man kann auch die Umkehrung nehmen. Nämlich "WHILE <>10". Diese Schleife läuft so lange, bis X ungleich 10 ist.
Daraus resultiert: Wird in einer Schleife UNTIL verwendet, wird die Schleife solange durchlaufen, bis (nach Booleschen Regeln) die Anweisung "wahr" wird. WHILE stellt praktisch das Gegenteil von UNTIL dar, die Schleife wird solange durchlaufen, wie die Anweisung "wahr" ist.
Um einen Austritt am Schleifenanfang zu erreichen, setzt man den Befehl UNTIL einfach hinter den Befehl DO. Diese Schleifenart nennt man abweisende Schleife, denn wenn der Befehl WHILE am Anfang der Schleife steht, wird, falls die Bedingung hinter WHILE erfüllt ist, der Schleifendurchlauf nicht mehr durchgeführt; wogegen bei unserer Beispielsschleife die Bedingung erst am Ende erfüllt sein muß. Diese Schleifenart nennt man abweisende Schleife. Der Befehl EXIT ermöglicht einen Ausstieg aus der Schleife zu einem bestimmten Zeitpunkt. Dieser erfolgt im Zusammenhang mit einer IF..THEN Abfrage, z.B. IFA$=50 THEN EXIT.EXIT unterbricht also die Schleife, um irgendwas in einer Variablen zu speichern. Nach EXIT verzweigt das Programm in die der Anweisung LOOP folgende Zeile. Die Struktur einer DO LOOP EXIT Schleife sieht folgendermaßen aus:

```
100 DO
110 ......
120 IF....THEN EXIT
130 ......
140 LOOP
```

Zum Schluß kommen wir zu der wohl schon allseits bekannten FOR... NEXT Schleife. Diese Schleifenart wird hauptsächlich als Zähl- oder Warteschleife angewendet. Auch zum Einlesen von Daten aus Datazeilen in den Speicher kann sie recht nützlich sein. Da die Überprüfung der Schleife am Ende erfolgt, wird eine Schleife mit FOR X =5TO5:NEXT nur einmal durchlaufen.
Halt! Eine Schleifenart hätten wir bald noch vergessen. Die Warteschleife mit GET. Das Basic 3.5 stellt uns hier den Befehl GETKEY zur Verfügung, das Programm wartet an dieser Stelle so lange, bis ein Zeichen über die Tastatur eingegeben wird. Erst dann fährt das Programm in seiner Abarbeitung fort.
**Fazit:** Durch die Schleifenbehandlung des Basic 3.5 wird eine Programmierung weitgehend ohne GOTO-Sprünge ermöglicht.

# NEUES AUS DER BÜCHERKISTE

## DAS ROM-LISTING FÜR DEN PLUS 4

Auf 260 Seiten stellt das Buch eine unerläßliche Informationsquelle für alle diejenigen dar, die sich näher mit dem Computer, insbesondere der Assemblerprogrammierung beschäftigen. Auch der normal sterbliche Programmierer findet hier eine Menge Informationen, die er zum Programmieren, bzw. zum Umschreiben von Programmen benötigt. Auf den ersten 6 Seiten findet der Leser die Speicherbelegung mit allen notwendigen Erklärungen dieser Speicherstellen. Auch befindet sich hier eine Auflistung der Speicherbereiche, die beim Einschalten der Grafikbereiche verschoben werden. Die restlichen 269 Seiten enthalten das komplette Rom-Listing als Assembler-Quellcode mit in deutscher Sprache kommentierten Codesequenzen.
Leider trübt der Preis von 59,– DM die Freude etwas. Zu beziehen ist dieses Buch über den Computer-Fachhandel oder über Buchhandlungen.

## DER COMMODORE 16 IN DER PRAXIS

Dieses Buch befaßt sich im einzelnen mit Zeitmessung, Grafik, Entwicklung und Gebrauch von Diagrammen, Musikprogrammierung und Büroarbeiten einschließlich einer Datenbank, eines einfachem Textverarbeitungssystems und einer Finanzverwaltung. Der Benutzer wird Schritt für Schritt durch die einzelnen Programmabschnitte der Kapitel geführt, wobei mit Erklärungen und Erläuterungen nicht gespart wird. Die einzelnen Module, die dann zu einem Programm zusammengesetzt werden, bestehen im Schnitt aus ca. 10 bis 30 Zeilen und sind somit leicht übersehbar. Auch wird jedesmal der genaue Ablauf des Moduls erklärt. Durch die sehr genauen Erläuterungen ist dieses Buch für den Anfänger empfehlenswert, wobei er noch zwei Fliegen mit einer Klappe schlägt. Zum ersten wird er Schritt für Schritt in die Programmierung des C16 eingeführt, zum anderen erhält er dadurch einige recht interessante Programme für seinen persönlichen Gebrauch. Das Buch kostet ca. 20,– DM.

## DAS GROSSE C-16-BUCH

„Alles, was Sie schon immer über den C-16 wissen wollten." So lautet der Untertitel dieses neuen Buches über den C-16 und seine Brüder, den C-116 und den Plus 4. Auf ca. 215 Seiten behandelt das Buch alle Themen, die den C-16-Programmierer schon lange interessieren.
Seien es die Grafikmöglichkeiten des eingebauten TED-Chips, oder die Soundprogrammierung in Basic und Maschinensprache. Auch wird ein Einführungskurs in die Maschinensprache dem Leser nicht vorenthalten. Die einzelnen Kapitel behandeln die verschiedenen Themen ausführlich und sind mit guten und ausreichenden Erklärungen versehen. Da der C-16 einen eingebauten Maschinensprachemonitor besitzt, war es möglich, fast alle in diesem Buch beschriebenen Beispiele in Basic und Maschinensprache darzustellen. Außer den vielfältigsten Beispielprogrammen aus allen Bereichen, wie Grafik, Sound und den verschiedensten Hilfsprogrammen, enthält das Buch noch verschiedene Listen, wie sie jeder Programmierer einmal braucht, um sich das Programmieren zu erleichtern. Etwa einen Befehlssatz des Prozessors 7501, einen Speicherplan (Memorymap), die ausführlich erläuterten Kernal-Routinen (mit Beispielen) und auch noch eine Vergleichstabelle der C-64 und C-16-Speicherstellen. Auch eine Tokenliste der wichtigsten Basic-Schlüsselwörter fehlt nicht. Durch seine knappe aber präzise Ausdrucksweise ist es dem Autor gelungen, alles, oder fast alles, was den C-16-Freak interessiert, in diesem Buch unterzubringen.

Das Buch, mittlerweile schon in seiner 2. Auflage erschienen, kann allen C-16-, C-116- und Plus-4-Besitzern nur wärmstens empfohlen werden.

*B. Welte*

**Info: Das große C-16-Buch.
Ca. 215 Seiten. Erhältlich bei Kingsoft, Schnackebusch 4, 5106 Roetgen.
Preis: 29,– DM**

## EFFEKTIV UND KREATIV MIT DEM PLUS 4

Mit dem Untertitel „Ein Wegweiser zum optimalen Einsatz" ist dieses Buch speziell auf den Plus 4 zugeschnitten, bzw. auf die eingebaute Software bezogen.
Das Buch versucht, nach einer ausführlichen Einleitung, über den Sinn und Zweck einer Text-, Datenverarbeitung, Tabellenkalkulation und eines Grafikpakets, eben die eingebaute Software, Tips für eine optimale Nutzung zu geben.
Nach einer ausführlichen Erklärung der Unterschiede zwischen deutscher und englischer Ausgabe des Plus 4 wird auf über 10 Seiten die Installation eines Druckers an den Plus 4 beschrieben. Auch die Installation eines Fremddruckers wird ausführlich behandelt. Allerdings ist hier die Hausmarke (Epsondrukker mit Data-Becker-Interface) Bezugspunkt. Die folgenden Seiten sind reserviert für das Erstellen einer Rechnung mit der Plus 4-Textverarbeitung, und beziehen sich hauptsächlich auf die Erstellung einer Druckermaske. Allerdings in allen Variationen,mit ausführlicher Erläuterung der Druckersteuerzeichen und mit vielen Ausdruckbeispielen in allen Größen angereichert. Weitere Kapitel beschäftigen sich mit der Plus 4-Tabellenkalkulation und der Dateiverwaltung. Auch die Grafik wird in einem Kapitel behandelt. Die letzten Kapitel des Buches beschäftigen sich mit verschiedenen Hilfsprogrammen, von der formatierten Eingabe bis zum Joystick-Painter. *Bernd Welte*

**Info: Froitzheim/Kausmann, Effektiv & Kreativ mit dem Plus 4, Data-Becker, Düsseldorf, ISBN 3-89011-073-8, Preis: 49,– DM**

## C16 TIPS & TRICKS

Hier stellen wir Ihnen ein neues Buch aus der 'Roten Reihe' der Data Becker-Bücher vor. Der Titel läßt auf eine Fülle von Informationen hoffen, man wird an das fast legendäre 'Tips & Tricks zum C64' erinnert. Das C16-Buch ist jedoch nur halb so dick, dafür ist die Schrift etwas größer ausgefallen.
Nun ist der C16 als Einsteigercomputer gedacht und dieses Buch richtet sich auch an diese Gruppe. Wer sich jedoch tiefgreifende Einsichten über die besonderen Tricks des C16 erhofft, ist weiterhin auf das Handbuch und andere Literatur angewiesen. Für den, der aber erst beginnt, die Welt der Computer zu erobern, bietet das Buch doch einiges Lernenswertes.
Sehr lobenswert ist die Methode, mit der die unleserlichen Steuerzeichen im Programmlisting vermieden werden.
Während Data Welt, die Tochterzeitschrift aus gleichem Hause, nach wie vor mit großer Hartnäckigkeit die reversen Ratekästchen in ihren Basiclistings abdruckt, hat sich der Autor dieses Buches zu der Methode entschlossen, die bei der COMMODORE WELT schon von Beginn an verwendet wird. Die Steuercodes werden durch eine Stringvariable ersetzt, welche dann im weiteren Programm verwendet wird (z.B. CL$=CHR$(147)).
Ebenfalls gut ist die Beschreibung der Programme. Die einzelnen Besonderheiten der Programmierung werden erklärt.
Das Kernstück des Buches bildet ein 'Textverarbeitungs- und Dateiverwaltungsprogramm', mit dem man sage und schreibe eine DIN A4-Seite bearbeiten (66 Zeilen zu 74 Zeichen) und darin Begriffe ordnen




und suchen kann. Dieses Programm ist den Verhältnissen des C 16 nicht angepaßt, da die Relation zwischen Programmlänge und Textlänge zu Ungunsten des Textes geht. Aber man will ja eigentlich daraus lernen.
Die sechs wichtigsten Programmiertricks sind nocheinmal extra ausgeführt.
Enttäuschend ist, daß die im Vorwort versprochenen wichtigsten Zeropage-Adressen sich in der Angabe von 3 Bereichen erschöpft, in denen Maschinenprogramme untergebracht werden und auch die angesagte Bedienungsanleitung des Monitors nicht wesentlich über das Handbuch hinausgeht.
Einige Angaben erwecken den Eindruck, als hätte der Autor sich nur zum Zwecke des Buchschreibens mit dem C 16 befaßt. Sonst wüßte er, daß man mit dem CHAR-Befehl sehr wohl den Cursor possitionieren kann, ohne dabei einen String auszugeben, indem man nämlich einen Leerstring angibt. Das gleiche gilt für den Tip, alle Tasten auf Dauerfunktion zu schalten, indem man POKE344,128 eingibt. Dies ist nämlich der Einschaltzustand!
Wer ein Buch zu seinem C 16 sucht, sollte sich das, was er kaufen will, genau ansehen und sich fragen, ob es ihm mehr bringt als das was in entsprechenden Zeitschriften abgedruckt ist. Bei diesem Buch ist das jedenfalls nur bedingt so.

**Preis:**
pro Band 29,– DM

**Technisches:**
201 Seiten – Format 14,5x21 cm, gebunden, auf Mattpapier gedruckt

## BASIC-WEGWEISER FÜR DEN COMMODORE 116, 16, Plus/4

In der Reihe der BASIC-Wegweiser wurde nun auch der C16 aufgenommen. Dieses, unter den C-16-Büchern ungewöhnlich umfangreiche, Buch, beginnt im ersten Viertel mit einer ausführlichen Einführung in aktuelles Grundlagenwissen zur Datenverarbeitung bzw. Informatik.
Dieser Teil ist unabhängig vom eigentlichen C16-Teil und kann getrost übersprungen werden, wenn man nicht sein Allgemeinwissen in punkto Computer aufbessern will.
Für den absoluten Computerneuling ist diese Vorgehensweise empfehlenswert, da er sonst vielleicht von der Fülle der Information, die mit seinem C16 direkt nichts zu tun hat, abgeschreckt werden könnte. Im zweiten Kapitel geht es dann über ca. 40 Seiten ausführlich um das 3.5er Basic des C 16, mit einer gut erklärten Liste der BASIC-Befehle. Hier wird auch auf die Bedienung der Tastatur und der Peripherie eingegangen sowie ein Vergleich des BASIC 3.5 zu BASIC 2.0 und BASIC 4.0 geliefert.
Das dritte Kapitel enthält einen 180 Seiten starken BASIC-Programmierkurs, der ganz auf den C16 abgestimmt ist. Anhand von vielen Programmen werden wichtige Programmstrukturen erklärt und die Grundlagen zur Planung und Entwicklung von Programmen dargelegt.
Es wird ausführlich auf Verarbeitung von Text und Tabellen sowie auf das Suchen und Sortieren von Daten eingegangen. Auch die Verwendung der Graphik und der Musik in BASIC-Programmen kommt nicht zu kurz.
Etwas störend ist nur die permanente Verwendung des LET-Befehls, der völlig unnötig ist. Statt 'LET A=B' braucht man einfach nur 'A=B' zu schreiben. Auch wird überhaupt nicht auf den Kassetten-Betrieb eingegangen, so daß der Leser, der keine Diskettenstation besitzt, die entsprechenden Programme umschreiben muß. Dies sollte jedoch mit Hilfe des Handbuches leicht möglich sein.
Das Buch eignet sich gut für den, der ernsthaft das Programmieren in BASIC erlernen will. Es wirkt etwas schulbuchhaft, was durch die lobenswerten grauen Kästchen, in denen wichtige Informationen hervorgehoben werden, noch verstärkt wird.
Keine Information wird der finden, welcher nach hardwarespezifischen Tricks zum C 16, wie die Zeropagebelegung und SYS-Adressen sowie den Einsatz von SHAPES sucht. Also nichts für Spieler, sondern nur für 'ernsthafte' BASIC-Programmierer.

**Untertitel:**
Datenverarbeitung mit BASIC 3.5

**Autor:**
Ekkehard Kaier
ISBN 3-528-04337-7

**Verlag:**
Vieweg Verlag
Postfach 5829
6200 Wiesbaden

**Bezugsquellen:**
Buchhandlungen – Computershops

**Technisches:**
304 Seiten – Format 16x23 cm, 87 Programmbeispiele, 116 Abbildungen, 40 Programmablaufpläne und Struktogramme, gebunden, auf Mattpapier gedruckt.
Die Programme sind auch auf Diskette zu 48,– DM erhältlich.

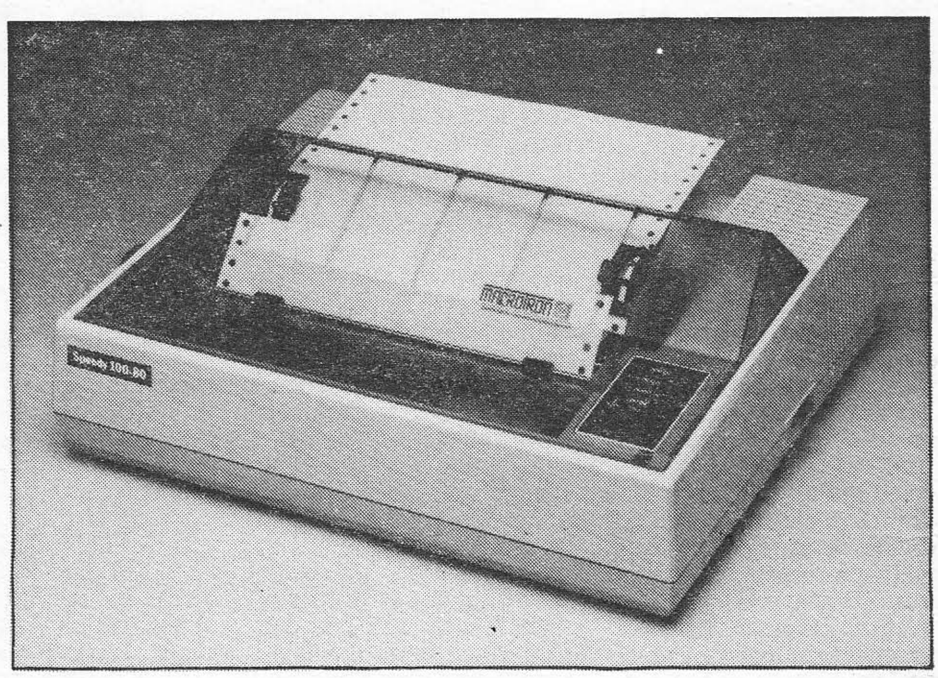
# LÜCKE GESCHLOSSEN: TEXTVER-ARBEITUNG AUCH FÜR C16!

Der private Computeranwender, der des Spielens überdrüssig geworden ist, sieht sich über kurz oder lang nach anderen Aufgaben um, die sein Computer übernehmen kann. Zwangsläufig zieht er außer Dateien auch eine Textverarbeitung in Betracht. Für die meisten Homecomputer, wie C-64 und 128 PC, gibt es diese Programme schon in sehr großer Zahl, so daß man hier schon fast die Übersicht verliert. Anders sieht es bei den "Bonsai-Computern" (C-16/C116) von Commodore aus.
Eine Ausnahme bietet die doch etwas magere Textverarbeitung des Plus 4. Für den C-16/C116 gibt es, soweit bekannt, nur eine Textverarbeitung der Firma Kingsoft.



## STECKMODUL SCHAFFT SPEICHERPLATZ

Die, da sie in den Speicher geladen wird, läßt nicht mehr viel Speicherplatz für den Text übrig. Diese Lücke schließt nun das neue Textverarbeitungssystem "Script/Plus" von Commodore. Als Modul für den Expansionsport belegt es keinen Speicherplatz und kann sich doch mit vielen hochwertigen Textverarbeitungen messen, die für andere Computer angeboten werden.
Nach dem Einschalten und Starten des Moduls mit der F2-Taste ist derjenige, der das Programm Easy-Script kennt, doch etwas überrascht, denn das Titelbild ist zu 90% mit diesem identisch. Kein Wunder, kommt es doch aus dem gleichen Softwarehaus.
Nach dem Einschalten und vor Eingabe des Textes muß der Benutzer nun noch einige Grunddaten eingeben. Die Anzahl der Zeichen pro Zeile geht von 40 – 240 Zeichen pro Zeile.




**Ein preisgünstiger Drucker für den C16/Plus 4: der Speedy**

Man sollte die angegebene Zeichenzahl(40) beibehalten, da sonst der Bildschirm seitlich auswandert und man dann leicht die Übersicht verliert. Die höhere Zeichenanzahl sollte man bei der Erstellung von Tabellen anwenden, da man hier sowieso mit der Tabfunktion arbeitet, um die einzelnen Punkte wiederzufinden.
Als nächstes wählt man zwischen Disketten- und Kassettenspeicherung aus. Auch die Druckerauswahl wird in diesem Menüpunkt vorgenommen. Hier stehen schon Drucker zur Verfügung. Nach Eingabe aller Daten gelangt man in den Editier-Modus. Jetzt kann man den gewünschten Text eingeben. Über das Zeilenende oder ähnliches braucht man sich keine Gedanken zu machen, denn das ist Aufgabe des Formatierens. Mit Hilfe des Trennungsvorschlages, der unter Nutzung der Escape und Commodore-Taste erfolgt, kann das Programm dann gegebenenfalls die Wörter trennen. (Anmerkung: Auf Seite 75 Abs. 3,11 des Handbuches steht es etwas anders!)
Hat man den Text vollständig eingegeben, kann man daran gehen, den Text zu formatieren. Dazu stehen dem Benutzer eine Vielzahl von Befehlen zur Verfügung. Diese werden immer an die Stelle im Text gesetzt, ab der sie wirksam werden sollen und beeinflussen den Ausdruck, bis sie entweder aufgehoben oder durch neue ersetzt werden. Vor jeder Formatierungseingabe muß die Tastenkombination >CONTROL 9< benutzt werden.

## UMFANGREICHER BEFEHLSSATZ

Folgende Befehle stehen zur Verfügung:
- rechten Rand setzen
- linken Rand setzen
- Papierlänge
- Textlänge
- Zeilenabstand
- Zentrieren
- Blocksatz
- rechtsbündiger Rand
- Zeilenvorschub

– Leerzeichen
– Rand lösen
– erzwungener Seitenumbruch
– Kommentarzeilen
– Schreibdichte
– Zeilendichte
– Seitenanfang
– Hoch- und Tiefschreibung
– Unterbrechung
und noch viele mehr.
Außer der vielfältigen Druckerausgabe können alle Texte natürlich auch auf Kassette oder Diskette abgespeichert und wieder geladen werden.
Als Besonderheit kann man hier betrachte, daß Text-File des Script/Plus und des Easy/Script untereinander ausgetauscht und auch bearbeitet werden, wenn man die unterschiedlichen Formatbefehle in Betracht zieht. Das trifft besonders auf den inversen Stern zu, der mit dem entsprechendem Programmbefehl geändert werden muß.
Eine weitere Besonderheit ist der eingebaute Rechner, der alle Grundrechenarten incl. Prozentrechnen im Text oder extern ermöglicht. Dadurch ist natürlich das Erstellen von Rechnungen und dergleichen zu einem Kinderspiel geworden.
Für Plus 4-Besitzer, die die etwas magere eingebaute Textverarbeitung leid sind, bietet Script/Plus etwas ganz Besonderes. Durch seinen großen Speicher bedingt, können zwei verschiedene Texte gleichzeitig im Speicher stehen und auch bearbeitet werden. Hierzu bietet das Programm den SWAP-Befehl an, der ein Umschalten zwischen den Texten ermöglicht.

## ZWEI TEXTE GLEICHZEITIG BEARBEITEN

Der wichtigste Teil eines Textverarbeitungsprogrammes ist natürlich die Be- und Verarbeitung von Texten. Auch in diesem Punkt hat Script/Plus einiges zu bieten:
– Löschen von größeren Textmengen
– Löschen von Buchstaben, Zeilen und Textblöcken
– Einfügen
– Verschieben
– Suchen
– Suchen und Austauschen
– Überspeichern
– Einbau einer Kopf- und/oder Fußzeile
– Unterstreichen und Fettdruck
– Form und Serienbriefe
– Tabulatoren und Zahlenbetrieb
– Globale Operationen:
* Suchen
* Suchen und Austauschen
* Ausgabe auf dem Bildschirm
* Ausdrucken
Script/Plus bietet dem Benutzer auch noch die verschiedensten Lösch- und Kopierfunktionen.
Auch können Textteile, die auf dem Drucker vergrößert dargestellt werden sollen, auf dem Bildschirm revers abgebildet werden. Such- und Austauschfunktionen sind natürlich auch in diesem Programm enthalten. Diese beziehen sich auf den Text im Hauptspeicher wie auch auf die Texte, die auf der Diskette gespeichert sind. Da wir schon einmal bei der Diskettenstation sind: Hier stehen dem Anwender alle Befehle der Floppy zur Verfügung. Das reicht vom Formatieren bis zum Einlesen und Ausdrucken der Directory.
Texte, die durch ihre Länge nicht mehr in den Speicher passen, können durch Verkettungsbefehle auf dem Drucker ausgegeben werden.
Auch besteht die Möglichkeit, mittels Datafiles Serienbriefe zu erstellen, wobei die Daten über spezielle Befehle an die richtige Stelle des Textes gesetzt werden.

## MIT SCRIPT/PLUS BEKOMMT DER DRUCKER FUTTER

Da Script/Plus als Modul im Expansionsport betrieben wird, kann es durch einen Befehl ein- und ausgeschaltet werden. Auch kann während des Arbeitens mit dem Programm ein Neustart durchgeführt werden, um z.B. die Zeichenzahl oder den Drucker zu wechseln, ohne daß der Text im Speicher verlorengeht.
Nach so viel Lob hält das Programm auch einen Tadel aus: das Handbuch. Nicht, daß es schlecht wäre. Nein. Aber, obwohl es mit ca. 150 Seiten die Fähigkeiten des Programms sehr gut und ausführlich erklärt, hat es doch einen schwerwiegenden Fehler, es ist nur in englisch erhältlich, was den deutschsprachigen Benutzer doch vor einige Probleme stellt.

## FAZIT

Das wohl beste Textverarbeitungsprogramm, das für die kleinen Commodore-Computer zur Verfügung steht. Auch Plus 4-Anwender sollten sich dieses Programm einmal ansehen, da es das eingebaute um Längen schlägt.
**Preis: 39,– DM**

*Bernd Welte*

# KEINE ANGST VOR DEM MONITOR

**Einer der Vorzüge der C16-Reihe (C16/C116 und Plus 4) ist zweifelsohne der eingebaute Maschinensprache-Monitor Tedmon. Im Anhang des Handbuches (Seite 218–225) sind die einzelnen Befehle erklärt. Der Anfänger benötigt aber doch noch einige Tips und Hinweise, wie er ihn sinnvoll verwenden kann. Dieser Artikel soll vor allem dazu dienen, einige Möglichkeiten aufzuzeigen, um dem Anfänger die Scheu vor Tedmon zu nehmen.**

Sicher denken viele, da sie ja vorerst mit Basic vollauf beschäftigt sind, daß Maschinensprache etwas sei, das nur Freaks verstehen. Diese sollten unbedingt weiterlesen.
Man kann den Tedmon auch von Basic aus nutzen. Will man vorhandene Maschinenroutinen in sein Programm einbauen, so kann man sie wie gewohnt mit:
LOAD "name",8,1 (RETURN)
einladen. Dabei verändern sich aber die Speicherstellen 43 – 46, die in Basic Programm-Anfang und -Ende anzeigen. Mit dem L-Befehl des Monitors kann man dies umgehen. Zunächst muß man den Tedmon mit
MONITOR (RETURN)
(oder m0 als Abkürzung) aufrufen. Die Registerinhalte, welche dann angezeigt werden, interessieren uns hier nicht. Mit
L"name",8 (RETURN)
(mit ,1 bei Kassette) wird das Maschinenprogramm geladen und nach
X (RETURN)
befindet man sich wieder im Basic, ohne daß die Speicherstellen 43–46 verändert sind. Das ganze kann man auch vom Basic-Programm aus machen. Dazu muß man sich der Tastaturpuffermethode bedienen. Zuerst wird der Text, den man beim Laden von Hand eingeben würde, mit CHAR-Anweisungen genau an den Stellen ausgegeben, wo man ihn in obigem Beispiel auch geschrieben hat. Dann wird der Cursor auf dem ersten Befehl (MONITOR) plaziert und in den Tastaturpuffer sooft die Zahl 13 (Kodezahl für RETURN) gepoket, wie man auch von Hand RETURN drücken würde. In die Speicherstelle 239 muß man noch die Anzahl der simulierten Tastendrucke poken. Dies hört sich für den Uneingeweihten vielleicht etwas kompliziert an. Deshalb hier ein kleines Beispielprogramm, welches dies alles macht.

```
10 rem ************************
20 rem **** maschinenprogramm
30 rem **** mit monitor laden
40 rem ************************
50 hk$=chr$( 34):cl$=chr$( 147)
60 input"programmname";p$:zn$="70":
goto 1000
70 printcl$"Programm ist eingeladen
"
80 end
990 rem * laderoutine ********
992 rem ** programmname in p$
994 rem ** zeilennummer in zn$
1000 char,0,0,cl$+"moN"
1010 char,0,6,"l"+hk$+p$+hk$+",8"
1020 char,0,9,"x"
1030 char,0,11,"goto"+zn$
1040 tp=1319:poke tp,19:fori=1to4:p
oke tp+i,13:next:poke239,5:end
```

Beispiel-Programm Monitor-Lader

### DER MONITORLADER

Falls Sie zu denen gehören, die überhaupt keine Maschinenroutinen benutzen (was sich am Ende dieses Artikels jedoch ändern wird), so überlegen Sie sich einmal, ob Sie nicht auf diese Weise HIRES-Bildschirme mit einer hübschen Grafik abspeichern und mit obiger Methode im Programm einblenden wollen.
Abspeichern eines HIRES-Bildschirms:

```
MONITOR                (RETURN)
S"HIRES",8,1800,
  4000                 (RETURN)
X                      (RETURN)
```

### DER ASSEMBLER IM TEDMON

Vielleicht wollen Sie es doch einmal versuchen, ein kleines Maschinenprogramm mit Hilfe des Assemblers einzugeben? Es ist ganz einfach. Zunächst einmal das Assemblerlisting:

FLASH WEG (stoppt das Blinken)

```
.05be  a0 00     ldy#$ 00
.05c0  b9 00 08  lda $  0800,y
.05c3  29 7f     and #$7f
.05c5  99 00 08  sta $0800,y
.05c8  b9 00 09  lda $0900,y
.05cb  29 7f     and #$7f
.05cd  99 00 09  sta $0900,y
.05d0  b9 00 0a  lda $0a00,y
.05d3  29 7f     and #$7f
.05d5  99 00 0a  sta $0a00,y
.05d8  b9 00 0b  lda $0b00,y
.05db  29 7f     and #$7f
.05dd  99 00 0b  sta $0b00,y
.05e0  c8        iny
.05e1  c0 00     cpy #$00
.05e3  d0 db     bne $05c0
.05e5  60        rts
```

Das sieht auf den ersten Blick recht kompliziert aus, aber in wenigen Minuten werden Sie es verstehen. Zunächst einmal müssen die sogenannten Mnemoniks, die Klartext-Assemblerbefehle, erklärt werden. Bei einem Maschinenprogramm läuft praktisch alles über die Register. Das A-Register ist einfach eine bestimmte Speicherstelle in der Zeropage, in der die entsprechenden Zahlenwerte abgelegt werden. Das gleiche gilt für das Y- und das X-Register. Am besten geben Sie mal das obige Assemblerprogramm ein, und nebenher lesen Sie die Erklärungen. Zum Eingeben eines Assemblerbefehls kann man an den Anfang der Zeile ein a oder einen Punkt schreiben:
.05be ldy #$00

Nach dem Punkt kommt zuerst die hexadezimale Adresse, bei der das Maschinenprogramm beginnen soll. 'ldy #$00' bedeutet, daß die Hex-Zahl $00 in das Y-Register geladen werden soll. Die nächste Adresse muß nicht mehr geschrieben werden, sie wird automatisch vorgegeben.
.05c0 lda $0800,y
'lda $0800,y': Der Wert der Speicherstelle $0800+(Inhalt des Y-Registers) wird in das A-Register geladen.(Angenommen, im Y-Register steht eine 5, so wird der Wert aus der Speicherstelle S0805 genommen.)
.05c3 and #$7f
Beim AND-Befehl wird immer der Inhalt des A-Registers mit der Zahl hinter AND 'undiert', das heißt, daß beim Ergebnis nur die Bits gleich eins sind, die in beiden Zahlen gleich eins sind. Durch die AND-Verknüpfung mit $7f=127 wird das 7. Bit der Zahl im A-Register gelöscht. Das Ergebnis steht nachher wieder im A-Register.
.05c5 sta $0800,y
Jetzt wird der Wert aus dem A-Register in die Speicherstelle $0800+ (Inhalt des Y-Registers) geschrieben. So, die Hauptarbeit ist erledigt. Wie man sieht, ist der größte Teil vom Rest eine Wiederholung mit leichten Änderungen. Fahren Sie jetzt einfach hoch mit dem Cursor auf 05c0 und ändern es in 05c8. Dann muß noch in der gleichen Zeile 0800 in 0900 geändert werden. Nach 2mal RETURN ändern Sie wieder 0800 in 0900 und drücken RETURN.

## SCHLUSS MIT DEM NERVÖSEN BLINKEN

Wenn Sie jetzt den Disassembler-Befehl
d 05be 05e5 (RETURN)
eingeben, sehen Sie schon den größten Teil des Programms. Nun haben Sie ja schon etwas Erfahrung im Überschreiben und können den Trick noch 2mal anwenden, bis Sie bei 05e0 angelangt sind. Geben Sie nun noch ein:

```
.05e0  iny
.05e1  cpy #$00
.05e3  bne $05c0
.05e5  rts
```

INY (increase y) erhöht den Inhalt des Y-Registers um 1, CPY (compare y) vergleicht ihn dann mit S00 und BNE (Branch if not equal) springt bei Nichterfüllung nach $05c0 zurück. Ist der Inhalt des Y-Registers gleich Null, so wird RTS (return from subroutine) ausgeführt und das Maschinenprogramm beendet. Der aufmerksame Leser wundert sich nun vielleicht, wie das Y-Register 0 werden soll, wenn immer 1 dazugezählt wird? Der größte Inhalt einer Speicherstelle eines 8-Bit-Computers ist 255 = 2 hoch 8 −1. Wird dazu nochmals 1 gezählt, so ergibt es wieder Null und in einer anderen Speicherstelle wird ein Bit gesetzt, welches anzeigt, daß hier ein Überlauf stattgefunden hat. Deshalb steht im Assemblerprogramm 4mal die gleiche Sequenz, um 1000 Byte zu verändern.

## SCHNELLER ALS BASIC?

Nun soll auch endlich erklärt werden, was dieses Programm leistet. Sie kennen ja sicher die Funktion der Help-Taste. Drückt man diese nach einem Programmfehler, so wird die fehlerhafte Stelle gelistet und blinkt, was einen auf die Dauer ziemlich nervös machen kann und beim Korrigieren ablenkt.
Mit diesem kleinen Maschinenprogramm ist damit Schluß. In den Speicherstellen von $0800 (dez. 2048) bis $0BFF (dez. 3071) befindet sich der Farbspeicher. Für jedes Zeichen auf dem Bildschirm kann die Farbe (16 Möglichkeiten = 4 Bits), die Luminanz (7 Möglichkeiten = 3 Bits) und FLASH ON oder FLASH OFF (2 Möglichkeiten = 1 Bit) festgelegt werden. Da jedes Byte 8 Bit hat, kann alles in diesem einen KByte untergebracht werden. Hierbei wird der FLASH-Modus durch ein gesetztes 7. Bit angezeigt. Wird der Speicherinhalt daher mit 127 undiert, so wird das Blinken abgestellt. Das Maschinenprogramm macht also das gleiche, wie dieses kleine Basic-Programm:

```
10 for i= 2048 to 3071:pokei,
peek(i)and127):next
```

Nur natürlich viel schneller. Der Platz für das Maschinenprogramm wurde am Ende des Funktionstastenspeichers gewählt, wo normalerweise diese 40 Byte leicht untergebracht werden können.
Um Übung zu bekommen,sollten Sie jetzt noch folgendes Assemblerprogramm abtippen: Dies macht das gleiche mit dem 7. Bit im Textspeicher, wodurch alle reversen Zeichen normal werden.
Mit ORA #$80 kann man übrigens genau den gegenteiligen Effekt erzielen. Wenn Sie die beiden Assemblerlistings vergleichen, so werden Sie große Ähnlichkeiten feststellen, die beim Eingeben bzw. Umwandeln des ersten in das zweite sehr hilfreich sind.
Versuchen Sie dann mal durch Veränderungen eigene Programme zu schreiben. Zum Beispiel kann man damit einen Bildschirminhalt verschieben, indem man die AND-Befehle wegläßt und hinter die STA-Befehle eine andere Adresse schreibt. Oder man läßt die STA- und die AND-Befehle weg, und schreibt die LDA-Befehle so: LDA #$01. Jetzt wird der Bildschirm mit lauter 'A' gefüllt.

REV WEG (macht revers zu normal)

```
.065e   a0 00      ldy #$00
.0660   b9 00 0c   lda $0c00,y
.0663   29 7f      and #$7f
.0665   99 00 0c   sta $0c00,y
.0668   b9 00 0d   lda $0d00,y
.066b   29 7f      and #$7f
.066d   99 00 0d   sta $0d00,y
.0670   b9 00 0e   lda $0e00,y
.0673   29 7f      and #$7f
.0675   99 00 0e   sta $0e00,y
.0678   b9 00 0f   lda $0f00,y
.067b   29 7f      and #$7f
.067d   99 00 0f   sta $0f00,y
.0680   c8         iny
.0681   c0 00      cpy #$00
.0683   d0 db      bne $0660
.0685   60         rts
```

Das Programm wird im Tedmon mit s"rev weg",8,065e,0686 (RETURN) abgespeichert und von Basic aus mit SYS1630 gestartet.

## ASSEMBLERLISTING AUF DEN DRUCKER

Will man ein Assemblerlisting ausdrucken, so muß man wie folgt vorgehen:

```
open1,4:cmdl       (RETURN)
monitor            (RETURN)
d 065e 0686        (RETURN)
x                  (RETURN)
print#1:close1     (RETURN)
```

Das gleiche geht natürlich auch mit dem M-Befehl.

## PLATZ FÜR MASCHINENPROGRAMME:

Hier ein paar Speicherbereiche, die beim C16/C116 für Maschinenprogramme benutzbar sind (wenn die angegebenen Funktionen nicht benutzt werden):

| | |
|---|---|
| RS232 Puffer | 03f7–0436 |
| Funktionstastensp. | 055f–05e6 |
| Sprachsynthesizerb. | 065e–06ee |
| Anwenderbereich | 00d8–00e8 |
| Kassettenpuffer | 0333–03f3 |

# TEXTE NACH BELIEBEN AUF DEM C16!

**Die Besitzer des Plus 4 haben ja ein Textverarbeitungssystem eingebaut. Für die C16er sieht es in dieser Beziehung ziemlich düster aus. Mit nur 12 KByte Speicher ist es allerdings auch schwierig, ein Textverarbeitungssystem zu schaffen, das allen Ansprüchen gerecht wird. Ein gutes Programm verbraucht soviel Speicher, daß nur noch Platz für ungefähr eine DIN A4-Seite zum Bearbeiten übrig ist.**



Die erste Schreibmaschine der Welt: Heute ist der Computer ihr Nachfolger.

Mit diesem Programm hat nun jeder endlich die Gelegenheit, auch lange Texte zu schreiben. Vor allem zwei Tricks ermöglichen dies:

**Erster Trick (Zeilen 430 – 490):**
Mit OPEN3,3 kann man den Bildschirm als Eingabekanal öffnen. Das bedeutet, daß damit die Zeichen im Bildschirmspeicher mit GET#3,A$ direkt (ohne Umrechnung) als ASCII-Zeichen gelesen werden können.

**Zweiter Trick (Zeilen 350 – 370):**
Mit OPEN1,0 kann die Tastatur als Eingabekanal gewählt werden. Dadurch wird es möglich, auf dem Bildschirm Veränderungen vorzunehmen.

**Programmablauf:**
Nach dem Eintippen, Abspeichern und Starten des Programms gibt man den Filenamen an, unter dem der Text abgespeichert werden soll. Dann kann man wählen, ob der Text auf der Diskettenstation gespeichert oder direkt auf dem Drukker ausgegeben werden soll. Gibt man Floppy + Drucker an, so wird der Text abgespeichert und gleichzeitig gedruckt. Dies ist eine Option, die wahrscheinlich kein anderes Textprogramm liefert. Auf diese Weise hat man einen Brief geschrieben und gleichzeitig abgespeichert.
Als nächste Wahlmöglichkeit gibt es Lesen oder Schreiben. Zuerst wird ja geschrieben. Nachdem die 2 gedrückt wurde, wird der Bildschirm gelöscht und oben links erscheint ein reverses Pfeil-nach-links-Zeichen. Jetzt kann Text eingegeben werden. Eine Zeile sollte nicht länger als 80 Zeichen sein und die ersten 10 – 15 Zeilen sollten mit SHIFT/RETURN abgeschlossen werden. Dabei bleibt jeweils ein Pfeilzeichen stehen, welches nachher beim Lesen vom Bildschirm ein Zeilenende anzeigt. An jeder beliebigen Stelle kann man korrigieren, indem man die f1-Taste drückt. Der Cursor blinkt und man kann ihn frei auf dem Bildschirm bewegen. Nachdem alle Fehler beseitigt sind, drückt man RETURN und der Cursor steht wieder an der Position zum Weiterschreiben. Da er nicht blinkt, wird der Pfeil in einen Stern umgewandelt. Wurde beim Korrigieren ein Pfeil überschrieben, so setzt man an das Zeilenende wieder einen Pfeil-nach-links (nicht revers). Hat man fast den ganzen Bildschirm vollgeschrieben, schließt man jede weitere Zeile mit RETURN ab. Ab jetzt wird immer die oberste Zeile ausgegeben (auf Drukker oder Floppy oder beidem), man hat also immer fast eine ganze Bildschirmseite Text, der korrigierbar ist. Wenn Sie zu den Menschen zählen, die

**VIELE KORREKTURMÖGLICHKEITEN**

beim Schreiben besonders viel korrigieren, so daß häufig Zeilen eingeschoben oder gelöscht werden müssen (was ja durch die ESC-Funktionen besonders gut geht), so sollten die Zeilen 350 – 370 durch:
350 open1,0:input#1,q$:close1:return
ersetzt werden. Jetzt springt der Cursor nach dem Korrigieren allerdings nicht mehr an die ursprüngliche Position, sondern muß nach der Veränderung und vor RETURN an seinen Platz gebracht werden. Ist der Text fertig, muß noch sooft RETURN gedrückt werden, bis der gesamte Text nach oben durchgeschoben und damit gesichert ist. Jetzt kann das Programm mit der RUN/STOP-Taste verlassen werden. Dabei werden die geöffneten Files regulär geschlossen (Zeilen 610 – 630). Die Textlänge beim Eingeben ist nur durch den Speicherplatz des Datenträgers begrenzt. Der Abspeichervorgang wird durch Veränderung der Rahmenfarbe angezeigt.
Beim Lesen des gespeicherten Textes geht man anfangs gleich vor. Beim Menüpunkt Lesen oder Schreiben gibt man 1 ein und der Text erscheint auf dem Bildschirm. Nach 15 Zeilen bleibt er ste-

```
10 rem = endlos-textprogramm ====
20 rem (p) 07/86   commodore welt
30 rem ===========================
40 rem (c) 06/86 r. schmid-fabian
50 rem c16/116 plus4  +1541/1531
60 rem ===========================
70 cl$=chr$( 147) :ho$=chr$( 19)
80 c1$=chr$( 157) :l2$=c1$+c1$
90 c3$=chr$( 29)  :r2$=c3$+c3$
100 c4$=chr$( 17)  :d2$=c4$+c4$
110 cz$=chr$( 95)  :cr$=chr$( 13)
120 ev$=chr$( 27)+"v"
130 f1$=chr$( 133) :key1,f1$
140 w$=chr$( 154)+chr$( 18)+chr$( 42)+
chr$( 146)+chr$( 157)+chr$( 144)
150 k$=chr$( 154)+chr$( 18)+cz$+chr$(
146)+chr$( 157)+chr$( 144) :printchr$(
14)
160 color0,2,4:color4,2,4:trap620
170 printcl$:input"programmname";p$
180 printcl$r2$d2$"floppy..........
...1"
190 print   r2$d2$"drucker.........
...2"
200 print   r2$d2$"drucker + floppy
...3"
210 geta$:if a$<"1" or a$>"3" then
210
220 a=val(a$):if a and 1 then f=1
230 if a and 2 then d=1:open4,4:rem
 drucker
240 printcl$r2$d2$"seq-File lesen..
...1"
250 print   r2$d2$"seq-File schreib
en 2"
260 geta$:if a$<"1" or a$>"2" then
260
270 if a$="1" then 540
280 if f then open8,8,2,p$+",s,w":r
em seq-file floppy
290 printcl$k$;
300 geta$:ifa$=""then300
310 printa$+k$;:if
a$=cr$ then gosub 400
320 if a$=f1$ then gosub350
330 goto300
340 rem tastatur als eingabe zum ko
rr.
350 open1,0:s=peek(202):z=peek(205)
360 input#1,q$:poke202,s:poke205,z:
sys55363
370 close1:printw$;:return
380 rem * 1 zeile vom bildsch. lese
n
390 rem * und auf drucker / floppy
400 z=peek(205):rem zeilenpos. curs
or
410 print ho$;:n=1:sc=1:p$=""
420 color4,2,3:rem rahmenfarbe
430 open3,3:rem bildschirm als eing
abe
440 get#3,a$:if a$=cz$ then a$=cr$
450 n=n+1:if n=40 then n=1:sc=sc+1
460 if f then print#8,a$;:rem flopp
y
470 if d then print#4,a$;:rem druck
er
480 if a$<>cr$ then 440
490 close3
500 printev$:if sc>1 then printev$
510 if z>1 then char,0,z-sc,""
520 color4,2,4:return:rem rahmenfar
be alt
530 rem** seq-file auf bildschirm *
***
540 open8,8,2,p$+",s,r":print cl$;
550 trap610
560 get#8,a$:ifa$=""then 560
570 if st=64 then 610
580 if a$=cr$ then printk$;:if peek
(214)>15 then if d then gosub 350:g
osub400
590 printa$;:goto560
600 rem* programmende ***********
610 if d then gosub350:printk$:fori
=1to15:gosub400:printk$:next
620 if f then print#8:close8:rem fl
oppy
630 if d then print#4:close4:rem dr
ucker
```

hen (Zeile 580, wenn die Druckeroption ausgewählt wurde), der Cursor blinkt und es können Veränderungen vorgenommen werden. Nach RETURN wird wieder jeweils die erste Zeile auf den Drucker geschickt. Man hat also viele Möglichkeiten zur Korrektur. Ist aller Text eingelesen, so blinkt der Cursor schwarz (ohne Pfeil) und die letzte Korrekturmöglichkeit ist angesagt. Nach RETURN wird der noch verbliebene Text abgeschickt. Will man ausdrucken, ohne zu korrigieren, so löscht man am besten vor Programmstart das 'gosub 350' in Zeile 580.

**Veränderungen beim Kassettenbetrieb:**

Um das Programm mit Datasette zu betreiben, müssen folgende Änderungen vorgenommen werden:

```
170 printcl$:input
    "Programmname"
    ;p$
200 print r2$d2$
    "drucker + kassette 3"
    240 print1Srd2$"
    Kassettenfile lesen  1"
250 print$r2$d2 "Kasset-
    tenfile schreiben 2"
280 if f then open8,1,1,
    p$:rem kassettenfile
    zum speichern öffnen
390 rem *und auf drucker/
    kassette*
530 rem**kassettenfile auf
    bildschirm***
540 open8,1,0,p$:print
    cl$;
585 if a$=cr$ then
    getkeyw$
620 if f then close8:rem
    kassette
```

# STEUERN & REGELN MIT DEM USERPORT

**Der Userport des Plus 4 bietet vielfältige Möglichkeiten, man muß sie nur zu nutzen wissen. Wir zeigen Ihnen hier an einigen kleinen Beispielen, wie mit ihm Steuer- und Regelungsaufgaben gelöst werden können.**

Eine angenehme Eigenschaft der Firma Commodore ist es, bis auf die Ausnahmen C16/C116, ihre Computer mit einem Userport auszustatten. Über den Userport, auf deutsch Anwenderschnittstelle, können die vielfältigsten Steuer- und Regelmaßnahmen durchgeführt werden. Die Beschaltung und Programmierung der Userporte für den VC-20 und C64 wurden schon in allen Richtungen durchleuchtet. Aber der Userport des Plus 4, um den es hier geht, gibt doch noch für viele etliche Rätsel auf. Da hier, im Gegensatz zum VC-20 bzw. C64, über den Plus 4 keine Angaben, auch von seiten der Firma Commodore, vorliegen, muß man schon selbst experimentieren. Es wurde schon behauptet, der Userport wäre mit dem des C64 identisch, aber leider stimmt dies nur für den mechanischen Teil. Doch dazu später. Aber Halt! Der Vorwurf an die Firma Commodore ist eigentlich ungerecht. Commodore hat die Informationen jedem Käufer des Plus 4 mitgegeben, im Handbuch.

Userport Plus 4 von hinten

1 2 3 4 5 6 7 8 9 10 11 12

A B C D E F H J K L M N

Wenn Sie nämlich die Seiten 253 – 256, das sind die Seiten mit den Schaltplänen, aufschlagen, finden Sie alles, was Sie brauchen. Aber wie das mit Schaltplänen so ist, die meisten verstehen sie nicht und so bleibt vieles im Dunkel des Computers verborgen. Schauen Sie sich einmal den Chip 7700 an. An seiner rechten Seite finden Sie die Leitungen F1 – F4. Sie enthalten außer den Angaben über die Pinbelegung auch die Adressen in hexadezimaler Darstellung. Und zwar folgendermaßen:

F1=$02
F2=$FD10–$FD1F
F3=$FD00–$FD0F
F4=$FDD0–$FDDF
F5=$FD30–$FD3F

Auch läßt sich aus dem Schaltplan die genaue Pinbelegung des Userports erkennen. Diese finden Sie an der rechten Seite des Chip 6529B. Und zwar sind das die acht Datenleitungen P0 – P7. Mit der Kombination von Daten-Ein- und Ausgabe an diesen Datenleitungen können vielfältige Steuer- und Kontrollaufgaben gelöst werden. Hierbei werden Datenwörter eingelesen, im Computer verarbeitet und anschließend wieder ausgegeben. Die genaue Userportbelegung finden Sie in Tabelle 1.

Um nun zu erkennen, wie die Datenleitungen arbeiten, haben wir eine kleine Schaltung abgedruckt, mit der alle möglichen Bitmuster an den 8 Datenleitungen sichtbar gemacht werden können. Der Aufbau der Schaltung, die relativ einfach gehalten wurde, erfolgt am besten auf einer Lochrasterplatine. Die Bauteile, die Sie dem Schaltplan entnehmen können, werden entweder mit Schaltdraht oder aber mit Hilfe der Fädeltechnik verbunden. Die Betriebsspannung für die Schaltung kann dem Computer an den Punkten 1 + 2 entnommen werden. Nur wenn Sie irgendwann einmal größere Lasten, wie Relais und dergleichen, schalten wollen, müssen Sie eine externe Stromversorgung anwenden. Da die Beschaltung etwas anders als bei den anderen Commodore-Computern ist, sollten Sie sich die Belegung in Tabelle 1 genau anschauen. Es geht hier etwas kreuz und quer zu. Um den Userport zu programmieren, benötigt man natürlich auch eine Adresse (Speicherstelle), in die man die entsprechenden Bitmuster poken kann, um dem Ausgabebaustein mitzuteilen, welche Ausgänge er nun zu schalten hat. Beim Plus 4 sind das die Adressen 64784 – 64799. Welche Sie nehmen, ist eigentlich egal, wir haben noch keinen Unterschied gefunden. Wenn Sie nun z.B. Poke 64784,x eingeben, erscheint an den Ausgängen das entsprechende Bitmuster, wobei x eine Zahl zwischen 0 und 255 sein kann.

Warum sich beim Poken in eine dieser Adressen auch alle anderen Adressen bis 64799 gleichermaßen ändern?

Vielleicht hat einer der Leser eine Erklärung und teilt uns diese einmal mit. Auch warum man nach Anlegen von Masse an einer der 8 Datenleitungen diesen Zustand mit Peek unter der gleichen Adresse abfragen kann, konnte noch nicht geklärt werden. Wie Sie sehen, Rätsel über Rätsel. Auch daß die Firma Commodore, im Gegensatz zur RS 232 Schnittstelle, nicht Genaues mitteilt, wird wohl immer schleierhaft bleiben.

## KLEINE PROGRAMM-BEISPIELE

Aber genug der Rätsel. Kommen wir nun zu den Listings, die ein paar Möglichkeiten aufzeigen, wie man den Userport programmiert. Listing 1 erzeugt ein Lauflicht oder eine Laufschrift, wobei auf dem Bildschirm die Ausgangsbelegung in allen möglichen Darstellungsformen angezeigt wird. (Von Dezimal über Hexadezimal bis Binär.)

Listing 2 zeigt die Steuerung einer Ampelanlage, wobei die Zeitabläufe in Schleifen erzeugt werden. Mit dem Programm aus Listing 3 können Sie mit Hilfe der Zahlentasten die Ausgänge ein- und ausschalten, wobei die Binärmuster auf dem Bildschirm sichtbar gemacht werden.

Listing 4 ist eine Erweiterung des Listings Nr. 1, wobei Sie jetzt mit dem

```
10 rem lauflicht plus4 userport =p4
20 rem (p) 03/86 commodore welt    =
30 rem ============================
40 rem (c) 03/86 by                =
50 rem bernd welte                 =
60 rem                             =
70 rem version 3.5  40z/ascii      =
80 rem plus 4 +1531/1541           =
90 rem ============================
100 z1$=chr$( 96)
110 forq=1to40:qa$=qa$+z1$:nextq
120 ddrb=64784
130 dimb( 8) :goto140
140 pokeddrb,255
150 printchr$( 147)
160 printtab( 6)"lauflicht plus 4  u
serport":print
170 print"led   dez. hex..........
..bin."
180 printqa$;:printchr$( 27)"t"
190 printchr$( 147)
200 forx=0to7
210 y=2^x
220 print:printtab( 1)x+1;:printtab(
6);:printusing"###";y;:printtab( 11)
hex$( y);
230 d=y
240 fori=8to1step-1
250 ifd/2=int( d/2)thenb( i)=0:else:b
( i)=1
260 d=int( d/2):nexti
270 fori=1to8:printtab( 17)b( i);:nex
ti:print
280 pokeddrb,y
290 fori=1to100:nexti:nextx
300 print:printtab( 12)"bitte eine t
aste"
310 getkeyq$
320 goto190
330 rem lauflicht userport ======p4
340 rem 000000 bytes memory      ===
350 rem 000000 bytes programm    ===
360 rem 000000 bytes variables   ===
370 rem 000000 bytes arrays      ===
380 rem 000000 bytes strings     ===
390 rem 000000 bytes fre (0)     ===
400 rem ===========================
```

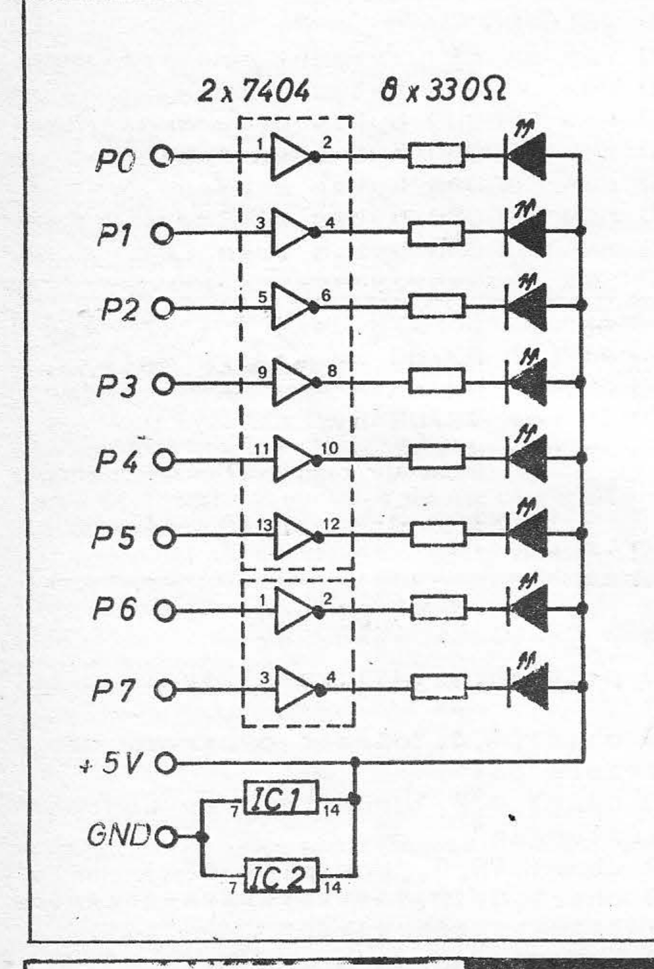
Joystick ein wenig rühren können, um so interessante Lichteffekte zu erzeugen. Wenn der Joystick in Ruhestellung ist, sind alle Leuchtdioden aus. Mit dem Feuerknopf können Sie alle wieder einschalten.
Wenn Sie nun mit den Listings ein wenig experimentieren, werden Sie vielleicht noch andere Möglichkeiten herausfinden. Sollten Sie zu neuen Erkenntnissen kommen, schreiben Sie uns. Denn auch wir wissen nicht alles. Vielleicht ist auch die Simulation einer Centronics-Schnittstelle möglich. Wer weiß.
Den Lötkolben angeheizt und frisch ans Werk.

```
10 rem ampel userp. plus/4 ======p4
20 rem (p)03/86 commodore welt     =
30 rem =============================
40 rem (c) 03/86 by                =
50 rem bernd welte                 =
60 rem                             =
70 rem version 3.5  40z/ascii      =
80 rem plus/4 + 1531/1541          =
90 rem =============================
100 printchr$( 147):c=64784:ti$="000
000"
110 pokec,2:fori=1to1000:next:rem g
elb
120 pokec,1:fori=1to2000:next:rem r
ot
130 pokec,3:fori=1to
400:next:rem rot/gelb
140 pokec,4:fori=1to5000:next:rem g
ruen
150 goto100
160 rem ampel userport ==========p4
170 rem 000000 bytes memory       ==
180 rem 000000 bytes programm     ==
190 rem 000000 bytes variables    ==
200 rem 000000 bytes arrays       ==
210 rem 000000 bytes strings      ==
220 rem 000000 bytes free (0)     ==
230 rem ===========================
```

```
10 rem userport plus/4 =========p4
20 rem (p) 03/86 commodore welt
30 rem ============================
40 rem (c) 12/85 by               =
50 rem bernd welte                =
60 rem                            =
70 rem version 3.5  40z/ascii     =
80 rem plus/4 + 1531/1541         =
90 rem ============================
100 rem userport plus 4
110 rem****************
120 printchr$( 147):rem**clear scree
n**
130 char1,4,4,"dieses programm demo
nstriert das"
140 char1,4,6,"arbeiten der userpor
t leitungen"
150 char1,15,8,"des plus 4"
160 char1,0,10,"*******************
*********************"
170 c=64784:pokec,0
180 char1,2,12,"die eingabe einer z
ahl wird durch die"
190 char1,8,14,"leds (1 bis 8) ange
zeigt"
200 char1,1,16,"led  dez.  hex.....
.....bin."
210 get r$:if r$=""then goto 210
220 r=val( r$)
230 char1,0,18,r$
```

*Tabelle 1*

| Pinbelegung | Userport Plus 4 | | |
|---|---|---|---|
| 1 | GND | | |
| 2 | +5V (100mA max.) | B | P0 |
| 3 | BRESET | C | RXD |
| 4 | P2 | D | RTS |
| 5 | P3 | E | DTR |
| 6 | P4 | F | P7 |
| 7 | P5 | H | DCD |
| 8 | RXC | J | P6 |
| 9 | ATN | K | P1 |
| 10 | 9V | L | DSR |
| 11 | 9V | M | TXD |
| 12 | GND | N | GND |




```
240 if r>0 and r<9 then goto260
250 print:print:printtab( 15)"falsch
e eigabe" :getkeya$:run
260 b=2^r/2
270 x=a and not b or b and not a
280 printtab( 5) x;tab( 12) hex$( x);
290 d=x:fori=8to1step-1
300 ifd/2=int( d/2) thenb( i)=0:else:b
( i)=1
310 d=int( d/2) :nexti
320 fori=1to8:printtab( 17) b( i);:nex
ti
330 poke c,x
340 a=x
350 for v=1to 100:next
360 goto210
370 rem userport =================
```

```
10 rem joystick lauflicht =======p4
20 rem (p) 03/86 commodore welt
30 rem ===========================
40 rem (c) 12/85 by              =
50 rem bernd welte               =
60 rem                           =
70 rem version 3.5  40z/ascii    =
80 rem plus/4 + 1531/1541        =
90 rem ===========================
100 c=64784
110 do
120 ifjoy( 1)=0thenpokec,0
130 ifjoy( 1)=1thenpokec,1
140 ifjoy( 1)=2thenpokec,2
150 ifjoy( 1)=3thenpokec,4
160 ifjoy( 1)=4thenpokec,8
170 ifjoy( 1)=5thenpokec,16
180 ifjoy( 1)=6thenpokec,32
190 ifjoy( 1)=7thenpokec,64
200 ifjoy( 1)=8thenpokec,128
210 ifjoy( 1)=128thenpokec,255
220 loop
230 rem joystick lauflicht ========
240 rem 000000 bytes memory      ==
250 rem 000000 bytes programm    ==
260 rem 000000 bytes variables   ==
270 rem 000000 bytes arrays      ==
280 rem 000000 bytes strings     ==
290 rem 000000 bytes free (0)    ==
300 rem ==========================
```

# FUNKTIONSTASTEN-ANDERS BELEGT

**Durch die Möglichkeit, die Funktionstasten auf einfache Weise mit Befehlen und Befehlsfolgen zu belegen, bietet der C16/116 bzw. der Plus 4 phantastische Arbeitserleichterungen beim Programmieren. Aber auch vom Programm aus sind die Funktionstasten vielfach verwendbar. Hier sollen nun einige Tricks dargelegt werden, die die meisten Anfänger und so mancher Profi noch nicht kennen.**



**Die Funktionstasten ermöglichen leichteres Programmieren**

Im Handbuch ist sehr gut erklärt, wie man die Funktionstasten mit einfachen Befehlen belegt. Aus den Erklärungen geht jedoch nicht hervor, daß man auch ganze Folgen von Befehlen auf einer Taste legen kann. Zum Beispiel:
key 1,"for i=43 to 46:?i,peek(i) :next" + chr$(13)
Danach werden die Inhalte der Speicherzellen 43 bis 46 ausgedruckt. Eine solche Befehlskette kann bis zu 128 Zeichen lang sein. Dann allerdings ist kein Platz für die anderen Funktionstasten mehr übrig.

## 128 BYTE AUF EINER TASTE

Natürlich kann man bei der KEY-Definition nicht mehr als 88 Zeichen (einschließlich 'key 1,' auf einmal eingeben. Aber man hat die Möglichkeit, einen Teil zuvor in einer String-Variablen unterzubringen und mit '+a$' zu verketten.
Diesen Trick kann man auch anwenden, um im Programmablauf Zeilen zu ändern. Hierzu ein nützliches Beispiel: Angenommen, man will mathematische Funktionen grafisch darstellen und hat bereits ein Programm, bei dem die Funktion 'y=sin(x)+x' in Zeile 100 gezeichnet wird. Man will aber verschiedene Funktionen sehen, ohne das Programm anzuhalten und die Zeile zu ändern.
Dazu wird mit einer INPUT-Anweisung nach der neuen Funktion gefragt:
80 input"y=";f$:print"Bitte fl druecken":end
Die Funktionstaste 'f1' wird wie folgt belegt:
key 1,"100 y="+f$+chr$(13) +"goto 90"+chr$(13)
Dadurch wird das Programm in Zeile 80 angehalten, Zeile 100 eingefügt, und mit 'GOTO 90' in Zeile 90 wieder fortgesetzt.
Übrigens wird im Speicher 'CHR$(13)' nicht als 8 Byte sondern nur als 1 Byte (nämlich 13) abgelegt. Dies kann man sich mit dem Monitor leicht ansehen. Gehen Sie mal mit dem Befehl Monitor in den Tedmon (oder mit 'moN' abgekürzt) und geben ein:
m 055f 05e6
Jetzt wird Ihnen der Speicherinhalt aufgezeigt und rechts sehen Sie die Umsetzung in ASCII-Zeichen. Vielleicht fragen Sie sich dabei, woher der C16 weiß, wieweit die Belegung der einzelnen Tasten geht? In den Speicherstellen 1375–1382 steht die Anzahl der Byte/Taste. So kann man zum Beispiel den bereits belegten Platz im Funktionstastenspeicher erfahren, indem man eingibt (im Direktmodus):
a=1375:for i=0 to 7:b=b+peek (a+i):next:print b
Normal sind dies 50. Ändert man die Tastenbelegung jedoch mit den Abkürzungen (Handbuch S. 202, Abkürzung für HELP ist 'heL) ab, so werden nur noch 26 Byte gebraucht. Den Rest kann man für kleine Maschinenroutinen verwenden. Nun bleiben wir aber noch kurz im Tedmon und bei der Frage, woher der C16 weiß, welche Befehle zu welchen Tasten gehören. In der ersten Zeile sehen Sie (nachdem Sie den 'm'-Befehl von vorhin eingegeben haben) 8 HEX-Zahlen. Addieren Sie zur ersten einmal die zweite und überschreiben die zweite mit 0. Nach RETURN sind die entsprechenden Speicherzellen verändert und Sie sehen, nachdem Sie mit 'x' wieder ins Basic zurückgekehrt sind und 'KEY' eingegeben haben, daß unter 'key 1,...' die Befehle von key1 und key2 stehen und key2 nicht aufgelistet wird. Beim Addieren von Hexzahlen müssen Sie natürlich beachten, daß zum Beispiel 6+4 nicht 10, sondern A ergibt.

## ERSETZEN UND EINFÜGEN MIT EINER TASTE

Manchmal muß man ein Befehlswort (zum Beispiel DATA) oft schreiben, oder mehrmals einen Befehl durch einen anderen ersetzen. Dazu wird eine Taste mit genau der Tastenfolge belegt, die man oft ausführen muß. Dies geht auch mit der INST- bzw. DEL-Taste. Hierbei schreibt man bei der Tastenbelegung statt 'INST' 'CHR$(148)+' und statt 'DEL' 'CHR$(20)+'. Genauso kann man auch den Cursor nach 'CHRS (13)' auf die Spalte der Folgezeile positionieren, an der man weiterschreiben möchte. Ein Beispiel: Man will alle 'INPUT' durch 'GETKEY' ersetzen. Da GETKEY 1 Zeichen länger ist, belegt man f1 wie folgt:
key 1,chr$(148)+"getkey"+chr$(13)
Jetzt braucht man nur noch auf das I von INPUT zu fahren und f1 zu drücken.

# TIPS & TRICKS: RAMOS KLEIN ABER OHO

**Das Betriebssystem des Plus 4 oder C16/C116 mit 64k-Erweiterung im RAM!**

Welcher Programmierer hat sich nicht schon einmal gewünscht, das Betriebssystem oder Basic-ROM seines Rechners nach seinen Wünschen zu verändern? Ein anderer Zeichensatz, neue, erweiterte oder verbesserte Befehle, deutsche Fehlermeldungen, eine eigene Begrüßung, andere Bildschirmfarben und, und, und . . . Leider gestattet es bis heute noch kein ROM-Baustein, sich durch POKEs oder den Maschinensprache-Monitor verändern zu lassen. Für Besitzer eines Plus 4 oder eines 64K aufgerüsteten C16/C116 gibt es allerdings eine andere Möglichkeit, im Betriebssystem oder Basic-Interpreter "rumzufuhrwerken": Bei diesen Rechnern ist der gesamte Adreßraum des Prozessors mit RAM belegt. Das ROM überlagert die "obere Hälfte" des RAMs (von Adresse $8000 bis $FFF). Was liegt also näher, als den Inhalt des ROMs in das "darunterliegende" RAM mit den gleichen Adressen zu kopieren, wo es dann leicht verändert werden kann? Nun, genau dies tut das Programm Ramos ("RAM-Operating-System") zunächst auch – doch das genügt noch nicht: Wenn man die im RAM-Bereich liegenden und – nehmen wir an – veränderte Kopie des ROMs zum Laufen bringen will, muß man auf das RAM umschalten. Aber das Betriebssystem schaltet immer wieder auf das ROM zurück und landet schließlich wieder ganz im ROM-Bereich – alle Veränderungen wären wirkungslos!! Man muß also alle Stellen im Betriebssystem, an denen auf das ROM umgeschaltet wird, ausfindig machen (das ist mir nach mühsamer Suche gelungen!) und sie so verändern, daß der Rechner "im RAM bleibt": Dies geschieht durch die POKEs im zweiten Teil von Ramos.

Am Ende des Programms wird auf das RAM umgeschaltet und ein Reset ausgelöst (Deshalb: Ramos nach dem Eintippen erst abspeichern!!).

Das neue Betriebssystem meldet sich mit:

RAMOS 264 BASIC
V3.5 28661 BYTES FREI
FERTIG.

Nun kann man mit dem Monitor oder mit POKEs Änderungen am Betriebssystem vornehmen. Wichtige Adressen im ROM sind z.B.:

32972–32995
($80CC–$80E3):
Einschaltmeldung
33166–33665
($818E–$8381):
Basic-Schlüsselworte
(ASCII-Code; der letzte Buchstabe jedes Wortes ist 'geSHIFTet')
33667–33874
($8383–$8452):
Adressen der Basic-Routinen (minus 1!)
33905–34379
($8471–$864B):
Basic-Fehlermeldungen
(kodiert wie Basic-Worte!)
53248–54271
($D000–$D3FF):
Zeichensatz Groß/Grafik
54272–55295
($D400–$D7FF):
Zeichensatz Klein/Groß
57382–57641
($E026–$E129):
Tastaturdekodiertabellen
(4x je 65 Bytes des entsprechenden ASCII-Codes für Taste allein, mit SHIFT, mit COMMODORE und mit CONTROL)
57651–57666
($E133–$E142):
Farbtasten-Codes (ASCII)
der 16 Farbtasten
57667–57682

```
100 rem "RAMOS - RAM-Betriebssystem
. Nur fuer PLUS4 oder
110 rem "C16/116 mit 64k-Speicherer
weiterung !
120 rem "(C) April '86 by Ulrich Sc
haefer -----------------------
130 rem "BASIC-Ende herabsetzen ---
------------------------------
140 poke 54,128 : poke 56,128 : clr
150 rem "Teil 1: ROM ins RAM kopier
en ---------------------------
160 data a9,80,85,d1,a0,00,84,d0,b1
,d0,91,d0,c8,d0,f9,e6
170 data d1,a5,d1,c9,fd,d0,f1,a2,48
,bd,00,ff,9d,00,ff,e8
180 data d0,f7,60,78,8d,3f,ff,4c,f6
,ff
190 ad=dec("0340") : for a=0 to 41
: read he$
200 poke ad+a,dec(he$) : next a : s
ys ad
210 rem "Teil 2: Aenderungen im RAM
------------------------------
220 nop=234:ram=63 : rem "Konstante
n fuer POKEs
230 poke 51175, 0 : rem "Zeichenge
nerator im RAM (GRAPHIC AUS)
240 poke 62282, 0 : rem "Zeichenge
nerator im RAM (TED-Init.-Tab.)
250 poke 52870,nop : rem "
260 poke 52871,nop : rem "NOP's sta
```

($E143–$E152):
Farbcodetabelle: Voreinstellungen für jede Farb-Taste
62264–62289
($F338–$F351):
Defaultwerte für Bildschirmcontroller (Register 0 – 25)
"Ramos" ist Reset-fest, d.h. auch nach einem Reset noch aktiv. Wünscht man diese Eigenschaft nicht, so löscht man in Ramos die Zeile 290.
Wenn man im Betriebssystem noch bis zu 2K-Byte für eigene Program-

### HIN- UND HER-SCHALTEN MÖGLICH

me haben möchte und man auf den im RAM selbstdefinierbaren Zeichengenerator verzichten kann, so kann man durch Löschen der Zeilen 230 bis 270 das Betriebssystem dazu zwingen, den "alten" Zeichengenerator im ROM zu benutzen. Das RAM von 53248 bis 55295 ($D000 bis $D7FF) steht dann zur freien Verfügung!
Wer sich intensiv mit dem ROM seines Rechners befassen möchte, dem sei das ROM-Listing zum Plus 4 von Commodore (Sachbuchreihe Band 6, ISBN 3-89133-006-5)

### BRENNE GERNE EPROMS

empfohlen, das ebenso auch für den C16/C116 gilt, da Basic und Betriebssystem der beiden Rechner identisch sind.
Ramos eignet sich auch ideal zum Austesten neuer oder veränderter Betriebssysteme, die auf Eproms gebrannt werden sollen (zu diesem Zweck habe ich es ursprünglich auch entwikkelt). Ich bin gerne bereit, gegen einen Unkostenbeitrag Eproms (2x27128 für Basic- und Betriebssystem-ROM) zu brennen.

```
tt 'ORA 2042'
270 poke 52872,nop : rem "
280 poke 53179,ram : rem "RAM-Umsch
altung bei Bankswitch-Routine
290 poke 65527,ram : rem "RAM-Einsc
halten nach ReSet
300 poke 33074,ram : rem "CHRGET-Ro
utine
310 poke 33102,ram : rem "Byte aus
RAM holen
320 poke 62315,76  : rem "
330 poke 62316,137 : rem "JMP 62345
 statt RAM-Test
340 poke 62317,243 : rem "
350 poke  1288,  0 : rem "Flag fuer
 RAM-Test loeschen
360 rem "Teil 2a: (kann weggelassen
 werden) ---------------------
370 t$="ramos 264" : for a=1 to len
(t$)
380 poke 32974+a,asc(mid$(t$,a,1))
: next a
390 poke 33023,asc("i")
400 t$="fertig" : for a=1 to len(t$
)
410 poke 34418+a,asc(mid$(t$,a,1))
: next a
420 rem "Teil 3: Auf RAM umschalten
 und ReSet -------------------
430 sys ad+35
440 end
```

# ANALOG-UHR

Dieses Programm demonstriert eindrucksvoll die vielfältigen Möglichkeiten, mit Basic 3.5 schön („strukturierte" Programmierung) und schnell (Grafik-Befehle) zu programmieren.
Eine Pascal-ähnliche, übersichtliche Struktur wird durch die DO...LOOP-Schleife und das Einrükken der Befehlszeilen erreicht (der Trick: Statt des ersten Leerzeichens hinter der Zeilennummer ein Grafikzeichen mit SHIFT- oder Commodore-Taste eingeben, danach soviele Leerzeichen wie gewünscht. Beim Listen verschwindet das Grafikzeichen, die Leerzeichen bleiben!). Leider lassen sich die GOSUBs nicht durch mit Namen gekennzeichnete Prozeduren ersetzen, wie das z.B. in Simon's Basic möglich ist.
Zeitraubende Winkelberechnungen in Basic (für die Zeiger der Uhr) werden durch die Option, bei DRAW und LOCATE auch Winkel und Längen anzugeben, überflüssig. Daher ist auch die Darstellung eines Sekundenzeigers möglich.
Die Analoguhr läuft unverändert auf dem Plus 4, dem C16 und dem C116.

*Ulrich Schäfer*

```
100 rem analoguhr c16/116/plus4
110 rem (c) by ulrich schaefer
120   dim l(2),w(2),z(5)
130   l(0)=60:l(1)=84:l(2)=72
140   w(0)=30:w(1)=6 :w(2)=6
150   xm=160:ym=100:r=100
160   graphic 0,1
170   input "uhrzeit (hhmmss) ";ti$
180   graphic 1,1
190   circle
1,xm,ym,100,100,0,360,0,1
200   paint 1,0,0:paint 1,319,199
210   for a=30 to 360 step 30
220     draw 0,xm,ym to 95;a
230     draw 1 to 5;a
240   next
250   i=0:gosub 420
260   do
270     i=0
280     for a=0 to 2
290       m=1:gosub 480
300     next
310     i=3:gosub 420
320     for a=0 to 2
330       if z(a)=z(a+i) then 360
340       m=0:gosub 480
350       z(a)=z(a+i)
360     next
370     get a$
380   loop while a$=""
390   graphic 0,1
```

```
400 end
410    rem prozedur zeiten holen
420      z(i)=val(left$(ti$,2))
430      if z(i) > 11 then z(i)=z(i)
-12
440      z(i+1)=val(mid$(ti$,3,2))
450      z(i+2)=val(right$(ti$,2))
460    return
470    rem prozedur zeiger zeichnen
480      draw m,xm,ym to l(a);z(a)*w
(a)
490    return
```

# DEUTSCHE UMLAUTE FÜR DIE DREI C!

Mit diesem Programm kann man dem Plus 4, dem C 16 und dem C116 (mit und ohne 16- oder 64-K-Erweiterung) die deutschen Umlaute beibringen. Dabei bietet das Programm folgende Vorteile:

## AUTOMATISCH ANGEPASST

Es paßt sich automatisch an die verschiedenen Speicherbaustufen an, indem es den neudefinierten Zeichensatz immer direkt unter das Basic-Ende legt (dadurch geht ca. 1 KByte Basic-Speicher verloren).
Der Zeichengenerator bleibt – im Gegensatz zu vielen anderen Programmen dieser Art – auch bei der Rückkehr in den Ready-Modus und nach Fehlermeldungen aktiv!!
Der Zeichengenerator kann beliebig nach eigenen Wünschen verändert werden. Die Basicadresse des Zeichengenerators wird vom Programm ausgegeben ("NEUE MATRIX AB ADRESSE: ...").

*Ulrich Schäfer*

### Deutsche Tastatur

```
100 rem deutscher zeichensatz fuer
plus4/c16/c116 mit und
110 rem ohne ram-erweiterung (passt
 sich automatisch an !)
120 rem (c) 20.6.1986 by ulrich sch
aefer-------------------
130 a=peek(56)-3:rem basic-ende her
absetzen
140 if a/4 <> int(a/4) then a=a-1:g
oto 140
150 poke 56,a:poke 54,a:poke 55,0:p
oke 53,0:clr
160 a=256*peek(56):print "neue matr
ix ab adresse";a
170 poke 1177,62:rem fuer peek auf
rom schalten
180 for i=0 to 1023:poke a+i,peek(5
4272+i):next
190 rem ready.-routine aendern ----
-----------------------
200 for i=0 to 65:poke 1633+i,peek(
34438+i):next
210 for i=0 to 37:poke 1699+i,peek(
51145+i):next
220 poke 1177,63:rem peek wieder au
s ram
230 poke 1630,76:poke 1631,3:poke 1
632,135:rem jmp 34563
240 poke 1635,250:rem bmi 1630
250 poke 1728,41:poke 1729,251:rem
and 251
260 poke 1737,76:poke 1738,207:poke
 1739,134:rem jmp 34511
270 poke 768,97:poke 769,6:rem neue
r vektor auf 1633
280 rem ram-zeichengenerator aktivi
eren ------------------
290 poke 65298,peek(65298)and251:re
m matrix im ram
300 poke 65299,(peek(65299)and3)or
a/256:rem matrixadresse
310 poke 740,a/256:rem matrixadress
e fuer char-befehl
320 poke 2042,0
330 printchr$(8);:rem zeichensatzum
schaltung verhindern
340 rem deutsche zeichen definieren
 ---------------------
350 for i=1 to 7:read c,k:key k,chr
$(c+64)
360 for x=0 to 7:read b:poke a+8*c+
x,b:next
370 next i
380 data 127,8,0,56,108,120,108,108
,120,96:rem "ss
390 data 102,1,0,102,60,6,62,102,62
,0      :rem "ae
400 data 104,2,0,102,60,102,102,102
,60,0  :rem "oe
410 data 108,3,0,102,0,102,102,102,
62,0    :rem "ue
420 data 123,4,102,24,60,102,126,10
2,102,0:rem "Ae
430 data 124,5,102,60,102,102,102,1
02,60,0:rem "Oe
440 data 126,6,102,0,102,102,102,10
2,60,0 :rem "Ue
450 end
```

```
Kommentiertes Assemblerlisting zu 'RAMOS'
0340 A9 80    LDA #$80     ; Startadresse high
0342 85 D1    STA $D1      ; Zähler high
0344 A0 00    LDY #$00     ; Startadresse low
0346 84 D0    STY $D0      ; Zähler low
0348 B1 D0    LDA ($D0),Y  ; indirekt aus ROM lesen
034A 91 D0    STA ($D0),Y  ; in RAM abspeichern
034C C8       INY          ; Zähler low erhöhen
034D D0 F9    BNE $0348    ; noch nicht wieder =0 ?
034F E6 D1    INC $D1      ; Zähler high erhöhen
0351 A5 D1    LDA $D1      ; zum Vergleichen laden
0353 C9 FD    CMP #$FD     ; schon $FD00 (RAM-Ende) ?
0355 D0 F1    BNE $0348    ; Schleife, solange ≠ $FD
0357 A2 48    LDX #$48     ; X als Indexzähler LOW
0359 BD 00 FF LDA $FF00,X  ; aus ROM lesen bis $FFFF
035C 9D 00 FF STA $FF00,X  ; in RAM abspeichern
035F E8       INX          ; Indexzähler erhöhen
0360 D0 F7    BNE $0359    ; Schleife, solange ≠ 0
0362 60       RTS          ; zurück ins BASIC-Prg.
-------------------------
0363 78       SEI          ; Interrupt verhindern
0364 8D 3F FF STA $FF3F    ; auf RAM umschalten
0367 4C F6 FF JMP $FFF6    ; ReSet-Routine
```

```
Kommentiertes Assembler-Listing der 'ROMPEEK'-Funktion
(analog zur PEEK-Funktion im ROM ab $9DFA)
065E A5 15    LDA $15      ; Inhalt von
0660 48       PHA          ; $14/$15 auf
0661 A5 14    LDA $14      ; den Stack
0663 48       PHA          ; retten
0664 20 E4 9D JSR $9DE4    ; 16-Bit-Wert holen
0667 A0 00    LDY #$00     ; Index 0
0669 B1 14    LDA ($14),Y  ; aus ROM lesen
066B A8       TAY          ; in Y übergeben
066C 68       PLA          ; alten Inhalt
066D 85 14    STA $14      ; von $14/$15
066F 68       PLA          ; wiederher-
0670 85 15    STA $15      ; stellen
0672 4C 81 9A JMP $9A81    ; Y nach FAC wandeln
```

```
Rompeek

100 data a5,15,48,a5,14,48,20,e4
110 data 9d,a0,00,b1,14,a8,68,85
120 data 14,68,85,15,4c,81,9a,00
130 a=dec("065e")
140 for i=0 to 23
150 read d$
160 poke a+i,dec(d$)
170 next i
180 poke 1280,76
190 poke 1281,aand255
200 poke 1282,a/256
210 rem " peek(x) liest aus RAM
220 rem "  usr(x) liest aus ROM
230 rem "(C)
by Ulrich Schaefer
```

**Assembler- und Basiclisting**

# ROM-PEEK: ZUGRIFF AUF's ROM

Oft benötigt man in Basic-Programmen den Zugriff auf das Betriebssystem- oder Basic-ROM, um Tabellen, Befehlslisten oder den Zeichengenerator nutzen zu können. Die PEEK-Funktion greift immer – auch beim C16 und C116 mit 16K RAM – auf das RAM zu. Eine kleine "Befehlserweiterung" stellt zusätzlich zur RAM-PEEK-Funktion noch die ROM-PEEK-Funktion zur Verfügung: Der Einfachheit halber habe ich die User(Benutzer)-Funktion, eine Basic-Funktion, die man selbst programmieren kann, verwendet. Man braucht dazu lediglich den User-Vektor (Adresse 1281/2) auf die eigene Maschinenroutine umzuleiten. Meine ROM-PEEK-Funktion ist analog zur RAM-PEEK-Funktion im Basic-ROM (ab Adresse S9DFA) aufgebaut. Statt PEEK(X) ruft man die neue Funktion mit USR(X) auf. Der Zugriff erfolgt bei Adressen größergleich 32768 nun auf das ROM; bei Adressen kleiner als 32768 wird wie bei PEEK aus dem RAM gelesen. Eine weitere Möglichkeit, auf das ROM zuzugreifen, besteht darin, vor dem Zugriff mit PEEK den Befehl POKE 1177,62 zu geben. Alle folgenden PEEK-Zugriffe beziehen sich dann auf das ROM. Allerdings bewirkt dieser POKE-Befehl, daß auch andere Tätigkeiten des Basic-Interpreters sich auf das ROM beziehen. Deshalb sollte man möglichst bald nach dem Zugriff mit PEEK den Befehl POKE 1177,63 folgen lassen, damit der ursprüngliche Zustand wieder hergestellt wird.

*Ulrich Schäfer*

# TIPS & TRICKS: SCHNELL PROGRAMMIEREN MIT STENO-BASIC!

Wie Sie sicher bei der Lektüre des Handbuches festgestellt haben, kann man die meisten Basic-Befehle mit 2 – 3 Zeichen abkürzen (Handbuch Seite 202). Die Verwendung dieser Abkürzungen spart nicht nur sehr viel Zeit, sondern reduziert die Tippfehler beim Programmeintippen gewaltig. Das fehlerfreie Eingeben der Befehlsworte ist vor allem für die Besitzer des "Gummicomputers" C116 ein Problem, da man nie genau weiß, ob die Taste nun angesprochen hat oder nicht. Es ist einleuchtend, daß die Gefahr des Vertippens reduziert wird, wenn man von 5 oder 6 Zeichen nur 2 eintippen muß. Dieses Programm hilft beim Einüben dieser Abkürzungen. Die Zeit zum Üben holt man leicht wieder beim weiteren Programmieren herein.
Außerdem macht es Spaß, denn es ist gleichzeitig ein Reaktions- und Konzentrationsspiel. Der Programmaufbau ist relativ einfach und gut zu verstehen. Etwas Grundlegendes zu den sogenannten Tokens (Basic-Befehle) sei noch gesagt. Im Speicher stehen die Befehle als Zahlen größer 127. Findet der Basic-Interpreter eine solche Zahl, so sieht er in einer Liste nach, welchem Befehl sie entspricht und handelt entsprechend. Wie aber erkennt er die Abkürzungen? Wird ein Befehl abgekürzt eingegeben, so ist das letzte Zeichen geshiftet, zum Beispiel "INPUT#" kann "iN" geschrieben werden.
Durch das geshiftete Zeichen weiß der Interpreter, daß der Befehl hier zu Ende ist und sieht in seiner Liste im ROM nach, wo er ein Wort findet, das mit "IN" anfängt. Das erste mit "IN..." nimmt er dann. Daher gibt es auch keine Abkürzung für INPUT, da INPUT hinter INPUT# steht.

**Programmablauf:**
Zuerst werden alle Befehlsworte mit ihrer Codezahl (mit der sie im Speicher stehen) und ihrer Abkürzung aufgelistet. Danach fragt das Programm nach der Anzahl der Befehle, welche abgefragt werden sollen. Anschließend wird nach den Abkürzungen der zufällig bestimmten Befehlsworte gefragt.
Hierbei kommt es auf die Zeit zur Beantwortung an, da diese gemessen wird. Wurden alle abgefragt, erhält man eine Fehlerstatistik mit Korrektur und verbrauchter Zeit.
Folgende Abkürzungen stehen nicht im Handbuch:

| Befehl | Token | Abkürzung |
|---|---|---|
| ELSE | 213 | eL |
| EXIT | 237 | exI |
| HELP | 234 | heL |
| STEP | 169 | stE |
| USING | 251 | usI |

GETKEY hat keine richtige Abkürzung, da es ein zusammengesetztes Wort aus GET und KEY ist. Daher kann man es auch GET KEY schreiben und gEkE abkürzen.
Bevor Sie nun anfangen, das Programm einzutippen, sollten Sie Ihr Augenmerk auf die Zeile 650 lenken. Sie hat mit dem eigentlichen Programm nichts zu tun, sehr wohl aber damit, wie man Zeitverlust und Fehler vermeiden kann.
Fangen Sie daher beim Eintippen mit diesen 3 Zeilen an:

```
1 data ::::::::::::::::::::
2 key 1,"run650"+chr$
(13)+"list1"+chr$(13)
+"list650"+chr$(13)
650 a= 0:fori=0to18step
2:poke4103+i,12 8+i/2
+a/2:poke4103+i+1,44:
next:poke4122,58
```

In Zeile 1 stehen nach dem Leerzeichen hinter DATA 20 Doppelpunkte. Wenn Sie dieses Miniprogramm starten, wird die Funktionstaste "f1" neu belegt und in Zeile 650 werden die Doppelpunkte durch die Token ersetzt. Drücken Sie nun f1, so werden Zeile 1 und 650 gelistet. In 1 stehen bereits die Befehlsworte von Zeile 420. Nun muß die Zeile 1 zur Zeile 20 gemacht werden, in Zeile 650 a=0 zu a=20 und f1 gedrückt werden. Jetzt stehen die nächsten 10 Befehle in Zeile 1. Diese wird jetzt durch Ändern der Zeilennummer zu Zeile 40 und in 650 wird a=20 zu a=40 und so weiter und so fort...
Dies Verfahren spart Zeit und vermeidet Feh-

```
10 rem stenobasic-trainer ===116/16
20 rem (p) 7/86 commodore welt     =
30 rem =============================
40 rem (c) 7/86 cw-team            =
50 rem                             =
60 rem c116/c16/plus 4 (c64/vc20) =
70 rem =============================

80 cl$=chr$(147):fa$=chr$(130)
90 printchr$(14)
100 tm=125:rem anzahl der token (c6
4/vc20 tm=75)
110 dima$(tm),b$(tm),a%(tm):printcl
$
120 fori=0 to tm:reada$(i):next
130 fori=0 to tm:read x:s=s+x
140 if x=1thenb$(i)=a$(i):goto180
150 b$(i)=left$(a$(i),x-1)+chr$(asc
(mid$(a$(i),x,1))or128)
160 ifx=len(a$(i))thenb$(i)=a$(i)
170 ifx=9thenb$(i)="?"
180 printi+128,a$(i),b$(i)
190 next:if s<>293 then print"Fehle
r in den DATA-Zeilen":end:rem c64/v
c20:s<>168
200 rem ** abfragezyklus ********
210 printcl$:input"wieviel Token so
llen abgefragt werden ";tk
220 for i=1 to tk
230 q=int(rnd(ti)*tm):if a%(q)<>0 t
hen 230
240 printcl$"Wie ist die Abkuerzung
 von: ";a$(q)
250 z=ti:input ab$:zs=zs+ti-z:rem r
eaktionszeit
260 if ab$=b$(q) then print"richtig
!!":a%(q)=1:goto280
270 print"Falsch, sie heisst: "fa$b
$(q):a%(q)=2
280 getkeyw$:next
290 printcl$"Auswertung"
300 rem** auswertung *************
310 for i=0 to tm
320 if a%(i)=1 then r=r+1
330 if a%(i)=2 then f=f+1:print"Fal
sch war:",a$(i)fa$,b$(i)
340 next:for t=1to6000:next
350 printcl$"Sie haben ":print
360 print"Richtig geraten:  ",r
370 print"Falsch geraten:  ",f
380 print:print"verbrauchte Zeit:",
int(zs/60)" sec"
390 input"Noch einmal? j/n";w$:if w
$="j" then run
400 end
410 rem* datas token ****
420 data end,for,next,data,input#,i
nput,dim,read,let,goto
430 data run,if,restore,gosub,retur
n,rem,stop,on,wait,load
440 data save,verify,def,poke,print
#,print,cont,list,clr,cmd
450 data sys,open,close,get,new,tab
(,to,fn,spc(,then
460 data not,step,+,-,*,/,^,and,or,
>
470 data =,<,sgn,int,abs,usr,fre,po
s,sqr,rnd
480 data log,exp,cos,sin,tan,atn,pe
ek,len,str$,val
490 data asc,chr$,left$,right$,mid$
,go,rgr,rclr,rlum,joy
500 data rdot,dec,hex$,err$,instr,e
lse,resume,trap,tron,troff
510 data sound,vol,auto,pudef,graph
ic,paint,char,box,circle,gshape
520 data sshape,draw,locate,color,s
cnclr,scale,help,do,loop,exit
530 data directory,dsave,dload,head
er,scratch,collect,copy,rename,back
up,delete
540 data renumber,key,monitor,using
,until,while
550 rem ** datas zeichenanzahl **
560 data2,2,2,2,2,5,2,2,2,2,2,2,3,3
,3
570 data3,2,2,2,2,2,2,2,2,2,9,2,2,2
,2
580 data2,2,3,2,3,2,2,2,2,2,2,3,1,1
,1
590 data1,1,2,2,1,1,1,2,3,2,2,2,3,2
,2
600 data3,2,3,2,3,2,2,2,3,2,2,2,3,2
,2
610 data2,2,2,2,2,2,3,2,2,3,2,4,2,3
,4
620 data2,2,2,2,2,2,3,2,2,2,2,2,3,3
,2
630 data3,3,2,3,3,3,2,2,3,3,4,3,3,2
,3
640 data4,2,2,3,2,2
650 a= 0:fori=0to18step2:poke4103+i
,128+i/2+a/2:poke4103+i+1,44:next
660 rem stenobasictrainer====16/116
670 rem 2406 bytes program         =
680 rem ===========================
```

ler. Die Erhöhung um 20 geht solange, bis Sie einen ILLEGAL QUANTITY ERROR erhalten. Am Schluß die Zeilen 1 und 650 löschen und mit RENUMBER 420,10 die Zeilennummer richtigstellen und den Rest des Programms eingeben.

Zum Schluß noch ein Tip: Wenn Ihnen mitten beim Schreiben eines Befehlswortes einfällt, daß Sie es eigentlich mit 2 Buchstaben hätten abkürzen können, so shiften Sie halt den 3ten. Das wird auch verstanden.

# SUPER: VOKABEL-TRAINER

Nach dem Start des Programmes erscheint zunächst ein kleiner Vorspann und dann ein Menü. Die Menüfunktionen sind auf die Funktionstasten des C-16 gelegt. Die Anzahl der Vokabeln muß minimal 4 Stück betragen und das Maximum ist bei 120 Stück.
Die Menüfunktionen im Einzelnen:

## F1 – EINGABE

Wie es der Name schon sagt, handelt es sich hierbei um den Eingabebefehl des Programmes. Der Computer fragt Sie folgendes ab:
1. Um welche Sprache handelt es sich?
2. Wieviel Vokabeln wollen Sie eingeben?

Wenn Sie alles korrekt eingegeben haben, geht der Computer in den Eingabemodus. Abwechselnd fragt er Sie die deutsche und die fremdsprachige Bedeutung der Wörter ab. Dieser Vorgang wiederholt sich so oft, bis die bestimmte Anzahl voll ist.

## F2 – LERNEN

Mit dieser Funktion können Sie Ihre vorher eingegebenen oder geladenen Vokabeln lernen.
Der Computer fragt Sie nacheinander in zwei verschiedenen Testarten ab.

**Test Nr. 1:**
Der Computer fragt Sie, welche Vokabeln abgefragt werden sollen:
1. Fremdsprache (F2)
2. Deutsch (F1)

Wenn Sie auch dies beendet haben, müssen Sie die Anzahl, bzw. wieviel Durchgänge Sie machen wollen, eingeben. Nun können Sie anfangen zu lernen. Der Computer zeigt Ihnen ein fremdsprachiges bzw. deutsches Wort an, und Sie müssen die entsprechende Bedeutung eintippen. "TED" – der elektronische Schiedsrichter, bewertet dann Ihre Eingabe. Dieser Vorgang wiederholt sich, bis die eingegebene Anzahl erreicht ist.

**Test Nr. 2:**
Zunächst erhalten Sie (wie bei Test 1) eine kleine Beschreibung des Lehrvorganges und dann geben Sie wieder die Anzahl der Durchgänge ein.
Der Computer schreibt zwei fremdsprachige Wörter und ein deutsches Wort auf den Bildschirm. Die fremdsprachigen Wörter sind mit F1 und F2 gekennzeichnet.
Welches Wort gehört zu dem deutschen Wort? – geben Sie F1 oder F2 ein.
Am Ende erscheint Ihre Testnote und die Anzahl der falschen Vokabeln in Prozent.

## F3 – MEDIUM

Mit Medium können Sie:
1. Bei leerem Speicher vorher abgespeicherte Vokabeln laden.
2. Vorher mit F1 eingegebene Vokabeln abspeichern.

Wählen Sie nun F1 oder F2 . . .

**1. Load**
Der Computer fragt Sie nach dem Filenamen (z.B. Lektionsname) und der Anzahl der Vokabeln, um entsprechend Speicherplatz zu reservieren. Haben Sie noch alte Vokabeln im Speicher, so werden Sie gefragt, ob Sie diese löschen wollen (nur so kann der Computer laden).

**2. Save**
Um die gewünschten Vokabeln abzuspeichern, benötigt der Computer einen Filenamen (z.B. Lektionsname). **Wichtig:** Notieren Sie sich den Filenamen und die Anzahl der Vokabeln (für späteres Laden).

# EINSTELL-HILFE

Die Justierung des Wiedergabe-Tonkopfes ihrer Datasette ist sehr wichtig, denn es besteht ein Zusammenhang zwischen der Tonkopfeinstellung und der Qualität der wiedergegebenen Signale des Gerätes. Die meisten fehlerhaften Bänder mit Computerprogrammen, die in den Geschäften zurückgegeben werden, sind nicht auf schlecht hergestellte Bänder oder Programmaufzeichnungen zurückzuführen, in den meisten Fällen ist ein schlecht eingestellter Tonkopf schuld. Vor allem nach der Invasion der C-16-Computer als Set trat dieser Effekt bei den mitgelieferten Datasetten sehr häufig auf. Ein Einstellset von Robcom soll hier nun Abhilfe schaffen. Es besteht aus zwei Kassetten, von denen die eine das Programm mit dem Prüfton auf der Rückseite enthält und die andere mit einem Reinigungsband gefüllt ist. Außer den zwei Kassetten liegen dem Set noch ein Schraubendreher und eine genaue viersprachige Gebrauchsanleitung bei. Auch Deutsch ist hier vertreten. Nach dem Laden und Starten des Programmes wird mit dem auf der Rückseite befindlichen Prüfton der Tonkopf der Datasette eingestellt. Dies geht recht einfach. Man muß nur darauf achten, daß die waagerechten Streifen auf dem Bildschirm so schmal wie möglich bleiben. Andernfalls muß der Tonkopf mit dem beiliegenden Schraubendreher vorsichtig eingestellt werden.

```
10 goto 110
20 dim e$(a):dim d$(a)
30 return
40 rem supertest =========16/116/p4
50 rem (p) commodore welt         =
60 rem ============================
70 rem (c) by torsten stenzel     =
80 rem version 3.5  40z/ascii     =
90 rem c 16/116/p4 + cass/disk    =
100 rem ===========================
```

rem: bitte beachten: Großbuchstaben im Listing stellen die betreffende n Grafikzeichen auf diesen Tasten d ar !!

```
110 graphic 0,1
120 color 4,2,7
130 g=0
140 a$="AAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAA"
150 gosub 2120:g=g+1
160 a$="-----supertest der vokabelt
rainer-----":gosub 2120:g=g+1
170 a$="---------1986 by torsten ste
nzel--------":gosub 2120:g=g+1
180 a$="------------ist geladen-----
------------":gosub 2120:g=g+1
190 a$="EEEEEEEEEEEEEEEEEEEEEEEEEEEE
EEEEEEEEEEE"
200 vol 7
210 gosub 2120:for i=1 to 4 :s=7
220 for p=1 to 4
230 vol s:sound 1,596,9:sound 1,685
,9:sound 1,739,9:s=s-2
240 vol 2 :sound
1,596,2:sound 1,685,3:sound 1,739,3
250 next p
260 s=4
270 for p=1 to 4
280 sound 1,596,9:sound 1,704,9:sou
nd 1,770,9:vol s:s=s-1
290 sound 1,596,3:sound 1,704,3:sou
nd 3,770,1
300 next p:next i:for a=1 to 900:ne
xt a
310 clr
320 scnclr:r=0
330 char 1,17,20,"s u p e r t e s t
"
340 char 1,17,22,"(c) torsten stenz
el"
350 g=0:a$="menue":gosub 2120:a$="-
----------":gosub 2120
360 a$="eingabe(f1)":gosub 2120
370 a$="lernen(f2)":gosub 2120
380 a$="medium(f3)":gosub 2120
390 g=g+1:a$="press f1,f2 or f3":go
sub 2120
400 key 1,"Q":key 2,"W":key 3,"E"
410 get a$
420 if a$="Q" then 460
430 if a$="W" then 720
440 if a$="E" then 2240
450 goto 410
460 scnclr:g=0
470 if a>3 then goto 2750
480 char 1,17,20,"s u p e r t e s t
"
490 char 1,17,22,"(c) torsten stenz
el"
500 a$="funktion nr.1 -eingabe-":go
sub 2120
510 g=g+1:a$="fremdsprache:":gosub
2120
520 input f$
530 g=g+1:a$="anzahl der vokabeln(-
120 stc.):":gosub 2120
540 input a
550 if a>120 then 530
560 if a<4 then 530
570 gosub 20
580 scnclr:g=0
590 for q=1 to a:let b$=str$(q):a$=
a$+b$
600 char 1,17,20,"s u p e r t e s t
"
610 char 1,17,22,"(c) torsten stenz
el"
620 a$="vokabel nr.":let a$=a$+b$:g
=g+2:gosub 2120
630 a$="deutsch:":g=g+1:gosub 2120
640 input d$(q)
650 a$=f$+":":g=g+1:gosub 2120
660 input e$(q)
670 if g>12 then g=0:scnclr
680 next q
690 a$="-----------ende der eingabe
-----------":g=g+13:gosub 2120
700 for z=1 to 900:next z
710 goto 320
720 scnclr:if a<1 then gosub 2200:g
oto 320
730 g=0
740 char 1,17,20,"s u p e r t e s t
"
750 char 1,17,22,"(c) torsten stenz
el"
760 a$="funktion nr.2 -lernen-":gos
ub 2120:g=g+1
770 a$="test nr.1":gosub 2120:g=g+1
780 a$="wollen sie "+f$+"-deutsch(f
1)oder":gosub 2120
790 a$="(f2)deutsch-"+f$+" lernen ?
":gosub 2120
800 get p$
810 if p$="Q" then l=1:goto 840
```

```
820 if p$="W" then l=2:goto 840
830 goto 800
840 g=0:scnclr:g=g+2
850 if l=1 then a$=f$+"-deutsch":go
sub 2120
860 if l=2 then a$="deutsch-"+f$:go
sub 2120
870 a$="wieviele woerter w.sie lern
en:":g=g+2:gosub 2120:input u
880 if u<1 then 870
890 scnclr:g=2
900 a$="ich w.ihnen e.wort auf den
bildschirm":gosub 2120
910 a$="schreiben und s.muessen das
 jeweilige":gosub 2120
920 a$="fremdspr.oder deuts.wort da
zu eingeben":gosub 2120
930 a$="viel spass...":gosub 2120
940 for s=1 to 1200:next s
950 for w=1 to u:scnclr:g=2
960 color 1,7,4:char
1,0,0,"CCCCCCCCCCCCCCCtest-nr.iCCCC
CCCCCCCCCCC"
970 char 1,0,10,"DDDDDDDDDDDDDDDDDD
DDDDDDDDDDDDDDDDDDDDDD"
980 color 1,1,7
990 gosub 1710
1000 char 1,17,20,"s u p e r t e s
t"
1010 char 1,17,22,"(c) torsten sten
zel"
1020 z=int (rnd(1)*a)+1
1030 if u=z then 1020
1040 let u=z
1050 if z<1 then 1020
1060 if l=1 then a$=f$+":"+e$(z):go
sub 2120
1070 if l=2 then a$="deutsch:"+d$(z
):gosub 2120
1080 if l=1 then a$="deutsch:":g=g+
2:gosub 2120:input q$
1090 if l=2 then a$=f$+":":g=g+2:go
sub 2120:input q$
1100 if l=1 and q$=d$(z) then 1150
1110 if l=2 and q$=e$(z) then 1150
1120 if l=1 then 1190
1130 if l=2 then 1240
1140 stop
1150 vol 7:sound 1,700,25
1160 char 1,10,18,"====richtig":for
 p=1 to 800:next p:char 1,10,18,"..
........"
1170 scnclr:vol 0:next w
1180 goto 1270
1190 vol 8:gosub 1830
1200 for s=1000 to 700 step -10:sou
nd 1,s,1:next s
1210 a$="falsch,es war "+d$(z):g=g+
2:gosub 2120
1220 for s=1 to 900:next s:r=r+1:go
sub 1710
1230 scnclr:next w:g=10:goto 1270
1240 a$="falsch,es war "+e$(z):g=g+
2:gosub 2120
1250 for s=1 to 900:next s:r=r+1
1260 scnclr:next w
1270 a$="--------ok test nr.1 ist be
endet-------":gosub 2120:g=0
1280 for p=1 to 700:next p
1290 scnclr
1300 a$="funktion nr.2 -lernen-":go
sub 2120:g=g+1
1310 a$="test nr.2":gosub 2120:g=g+
1
1320 a$="ich schreibe ihnen ein deu
tsches":gosub 2120
1330 a$="u.zwei fremdspr.woerter au
f den":gosub 2120
1340 a$="bildschirm-sie muessen nun
 auf f1 o.f2":gosub 2120
1350 a$="tippen u.so d.richtige loe
sung finden.":gosub 2120
1360 a$="wiev.durchgaenge wol.sie m
achen:":g=g+1:gosub 2120:input
y:g=0
1370 if y<1 then 1360
1380 for h=1 to y
1390 g=0:scnclr:color 1,7,4
1400 char 1,0,0,"CCCCCCCCCCCCCCCtes
t-nr.iiCCCCCCCCCCCCCCC"
1410 char 1,0,11,"CCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCC"
1420 color 1,1,7
1430 gosub 1710
1440 z=int(rnd(1)*a)+1:if z<1 then
1440
1450 x=int(rnd(1)*a)+1:if x<1 then
1450
1460 let c=q
1470 if z=x then 1440
1480 g=g+2:a$=f$+":"+e$(z)+"(f1)":g
osub 2120
1490 char 1,17,20,"s u p e r t e s
t"
1500 char 1,17,22,"(c) torsten sten
zel"
1510 g=g+1:a$=f$+":"+e$(x)+"(f2)":g
osub 2120
1520 q=int(rnd(1)*2)+1:if q<1 then
1520
1530 if u=q then 1520
1540 let u=q
1550 if q=2 then a$="deutsch:"+d$(x
):g=g+2:gosub 2120:goto 1600
1560 if q=1 then a$="deutsch:"+d$(z
):g=g+2:gosub 2120:goto 1590
```

```
1570 if c=q then 1520
1580 goto 1520
1590 let c$=""
1600 if q=1 then let c$="f1":goto 1
620
1610 if q=2 then let c$="f2":goto 1
620
1620 get b$
1630 if b$="Q" then 1660
1640 if b$="W" then 1660
1650 goto 1620
1660 if q=1 and b$="Q" then 1910
1670 if q=2 and b$="W" then 1910
1680 vol 7:for s=500 to 1000 step 1
0:sound 1,s,1:next s
1690 gosub 1830:for p=1 to 700:next
 p
1700 goto 1880
1710 char 1,5,15,"U***I"
1720 char 1,5,13,"t e d"
1730 char 1,5,16,"GW WH"
1740 char 1,5,17,"T Q Y"
1750 char 1,5,18,"GJFKH"
1760 char 1,5,19,"J---K"
1770 char 1,5,14,"  I"
1780 char 1,5,20," N M"
1790 char 1,5,21,"O   P"
1800 char 1,5,22," ted "
1810 char 1,5,23,"====="
1820 return
1830 char 1,5,18,"GU-IH"
1840 char 1,5,16,"G   H"
1850 char 1,5,14,"  U"
1860 char 1,10,18,"===falsch"
1870 return
1880 gosub 1830
1890 r=r+1:g=g+1:a$="falsch,es war
"+c$:gosub 2120:for p=1 to 600:next
 p:next h
1900 goto 1940
1910 vol 7:sound 1,700,15:char 1,10
,18,"=====richtig":for p=1 to 700:n
ext p
1920 char 1,10,18,"..........."
1930 next h:goto 1940
1940 scnclr:g=4
1950 a$="----funktion nr.2 -lernen-
 beendet----":gosub 2120
1960 l=int(100/(y+u)*r)
1970 v$=str$(l)
1980 for p=1 to 500:next p
1990 a$="sie hatten "+v$+"% d.voka.
falsch":g=g+2:gosub 2120
2000 if l>5 and l<15 then n=2
2010 if l<5 then n=1
2020 if l>14 and l<30 then n=3
2030 if l>29 and l<50 then n=4
2040 if l>49 and l<60 then n=4
2050 if l>59 and l<70 then n=5
2060 if l>70 then n=6
2070 let n$=str$(n)
2080 a$="ihre testnote ist "+n$:g=g
+1:gosub 2120
2090 g=g+2:a$="ich hoffe ihre "+f$+
"-kenntnisse":gosub 2120
2100 a$="sind jetzt etwas besser...
":gosub 2120
2110 for p=1 to 1000:next p:goto 32
0
2120 f=0
2130 vol 7
2140 char 1,1,g,left$(a$,f)
2150 let b=len (a$)
2160 sound 3,800,1
2170 if f>b then g=g+1:vol 0:return
2180 f=f+1:if f>40 then f=0
2190 goto 2140
2200 g=1:a$="ich habe noch keine vo
kabeln":gosub 2120
2210 a$="im speicher......error...!
!":gosub 2120
2220 for t=1 to 600:next t
2230 return
2240 scnclr:g=0
2250 char 1,17,20,"s u p e r t e s
t"
2260 char 1,17,22,"(c) torsten sten
zel"
2270 a$="funktion nr.2 -medium-":go
sub 2120
2280 a$="wollen sie (f1)laden oder
(f2)saven ?":g=g+2:gosub 2120
2290 get b$
2300 if b$="W" then 2330
2310 if b$="Q" then 2510
2320 goto 2290
2330 if a<4 then scnclr:gosub 2200:
goto 320
2340 a$="saven-kas.in rekorder einl
egen !":gosub 2120:g=g+1
2350 a$="name des files:":gosub 212
0:input n$
2360 a$="notieren sie sich den file
name":gosub 2120
2370 a$="und d.anzahl der vokabeln-
":gosub 2120
2380 a$="(fuer spaeteres laden...!!
!)":gosub 2120
2390 b$=str$(a):a$="es sind..."+b$+
" stc.":gosub 2120
2400 open 1,1,1,n$:for g=1 to a
2410 print#1,e$(g):next g
2420 close 1
2430 open 1,1,1,n$:for g=1 to a
2440 print#1,d$(g):next g
2450 print#1,f$
```

```
2460 close 1
2470 scnclr:g=14
2480 a$="----------------ende save--
------------":gosub 2120
2490 for p=1 to 600:next p
2500 goto 320
2510 if a>2 then goto 2650
2520 g=g+1:a$="load-kas.in recorder
 einlegen":gosub 2120
2530 a$="anzahl der vokabeln:":gosu
b 2120:input a
2540 dim e$(a):dim d$(a)
2550 a$="name des files:":gosub 212
0:input n$
2560 open 1,1,0,n$:for g=1 to a
2570 input#1,e$(g):next g
2580 close 1
2590 open 1,1,0,n$:for g=1 to a
2600 input#1,d$(g):next g
2610 input#1,f$
2620 close 1
2630 scnclr:g=14:a$="--------------
ende load-----------------":gosub 212
0
2640 for p=1 to 600:next p:goto 320
2650 scnclr:g=2:a$="der vokabelspei
cher ist voll...":gosub 2120
2660 a$="wollen s.die alten vok.loe
schen(f1=ja)?":gosub 2120
2670 get a$
2680 if a$="Q" then 2710
2690 if a$="" then goto 2670
2700 goto 320
2710 a$="ok...ich loesche...":gosub
 2120:for p=1 to 600:next p
2720 clr
2730 scnclr
2740 goto 2510
2750 scnclr:g=2
2760 a$="der vokabelspeicher ist vo
ll,soll ich":gosub 2120
2770 a$="neu starten (f1=ja)?":gosu
b 2120
2780 get b$
2790 if b$="" then 2780
2800 if b$="Q" then clr:goto 10
2810 goto 320
2820 rem supertest =======16/116/p4
2830 rem 48383 bytes memory       ==
2840 rem 08199 bytes program      ==
2850 rem 00122 bytes variables    ==
2860 rem 00009 bytes strings      ==
2870 rem 00000 bytes arrays       ==
2880 rem 40053 bytes free         ==
2890 rem ========================
```

```
10 rem == minotaurus =============
20 rem (p)         commodore welt =
30 rem ===========================
40 rem (c)          cw-team       =
50 rem c16/116/plus 4             -
60 rem ===========================
70 c4$=chr$(17):c2$=chr$(145)
80 c3$=chr$(29):c1$=chr$(157)
90 trap 110:zb=1000
100 color1,1:color0,2,3:color4,1
110 graphic1,1
120 char,0,0,""
130 sshapem$,0,0,4,4:gshapem$,0,0,4
140 sshapep$,2,2,3,3
150 forz=1to 23:fors=0to39
160 x=int(rnd(1)*2):char,s,z,chr$(1
64+x)
170 next:next
180 draw,0,8 to 319,8
190 draw,0,192 to 319,192
200 x=155:y=194:gshapem$,x,y
210 ti$="000000"
220 getkeya$:gshapep$,x+1,y+1,3
230 if a$=c3$ thengshapem$,x,y,4:x=
x+4:gosub300:x=x+4:gshapem$,x,y,4
240 if a$=c1$ thengshapem$,x,y,4:x=
x-4:gosub300:x=x-4:gshapem$,x,y,4
250 if a$=c2$ thengshapem$,x,y,4:y=
y-4:gosub300:y=y-4:gshapem$,x,y,4
260 if a$=c4$ thengshapem$,x,y,4:y=
y+4:gosub300:y=y+4:gshapem$,x,y,4
270 ify=186then draw0,x,y+6 to x+4,
y+6
280 ify=2then draw0,x,y+6 to x+4,y+
6:goto330
290 goto220
300 gshapem$,x,y,4:gshapem$,x,y,4:r
eturn
310 goto110
320 rem * zeitauswertung *
330 zn=int(ti/60):if zb>zn then zb=
zn
340 paint,0,0
350 char,0,0,"neue zeit:":char,11,0
,mid$(str$(zn),2)
360 char,23,0,"beste zeit:":char,35
,0,mid$(str$(zb),2)
370 char,0,24,"noch ein spiel j/n":
getkeyf$
380 if f$="j" then goto 110
390 if f$<>"n" then goto 370
400 graphic0:end
```

```
10 rem fieberkurve ============c16
20 rem (p)        commodore welt
30 rem ==========================
40 rem (c)          by          =
50 rem paul saar                =
60 rem                          =
70 rem version 3.5  40z/ascii   =
80 rem c16/c116/plus 4 +1531/1541=
90 rem ==========================
100 graphic0,1
110 scnclr
120 print"  es wird behauptet , die
 commodore.......computer 16,116,pl
us 4 waeren tot."
130 print"  mit einfachen befehlen
des graphicbe-...reiches lassen sic
h die vorteile..."
140 print"..der vorgenannten comput
er nicht von.....der hand weisen !"
150 print"..auch 64 freaks muessen
zugeben , das....erweiterte basic 3
,5 wuerde auch "
160 print"  64 programmierern die a
rbeit stark......erleichtern."
170 print"..beispiel: erstellung ei
ner fieberkurve"
180 for t=1 to12000:nextt
190 scnclr
200 print"fieberkurve erstellen"
210 input"monat............";m$
220 input"name.............";n$
230 scnclr
240 graphic2,1
250 x=a+4:a=x+4
260 draw1,g,xto 320,x
270 ifx=> 160then x=0:goto290
280 goto250
290 y=b+4  :b=y+4
300 draw1,y,4 to y, 160
310 ify=> 320then goto 330
320 goto290
330 char1,5,19,"1234567890123456789
0123456789012345"
340 char1,1,18,"36,0"
350 char1,4,17,"1"
360 char1,4,16,"2"
370 char1,4,15,"3"
380 char1,4,14,"4"
390 char1,4,13,"5"
400 char1,4,12,"6"
410 char1,4,11,"7"
420 char1,4,10,"8"
430 char1,4,9,"9"
440 char1,1,8,"37,0"
450 char1,4,7,"1"
460 char1,4,6,"2"
470 char1,4,5,"3"
480 char1,4,4,"4"
490 char1,4,3,"5"
500 char1,4,2,"6"
510 char1,4,1,"7"
520 char1,10,1,m$
530 char1,20,1,n$
540 c=44
550 draw1,44,148
560 print"..........................
...................................
......."
570 print"..........................
............................."
580 input"  temperatureingabe in gr
ad celsius.......(36.0-37.7)  tag
1-31 ";b
590 vol8
600 ifb=<35.9 or b=>37.8then print"
..falsche eingabe ":sound1,800,10:g
oto560
610 ifb=36.0thena=148
620 ifb=36.1thena=140
630 ifb=36.2thena=132
640 ifb=36.3thena=124
650 ifb=36.4thena=116
660 ifb=36.5thena=108
670 ifb=36.6thena=100
680 ifb=36.7thena=92
690 ifb=36.8thena=84
700 ifb=36.9thena=76
710 ifb=37.0thena=68
720 ifb=37.1thena=60
730 ifb=37.2thena=52
740 ifb=37.3thena=44
750 ifb=37.4thena=36
760 ifb=37.5thena=28
770 ifb=37.6thena=20
780 ifb=37.7thena=12
790 drawtoc,a
800 c=c+8
810 ifc=>288 then goto 830
820 goto560
830 print "    eingaben beendet"
850 rem fieberkurve ============c16
860 rem 012277 bytes memory      ==
870 rem 002409 bytes programm    ==
880 rem 000000 bytes variables   ==
890 rem 000000 bytes arrays      ==
900 rem 000000 bytes strings     ==
910 rem 000000 bytes fre (0)     ==
920 rem ==========================
```

Dieses kleine Programm veranschalicht die Leistungsfähigkeit für Grafiken im Basic des 16/P4 Die Abfrage erfolgt zuerst nach Namen und Datum der entsprechenden Grafik, welches später in diese eingeschrieben wird. Daraufhin wird eine Tabelle erstellt, welche vertikal die Temperaturgrade und horizontal die Monatstage darstellt. Nun müssen für alle Tage die entsprechenden Temperaturen eingegeben werden, für die die Fieberkurve berechnet werden soll.

# BARACUDA

Mit diesem Adventure stelle ich Ihnen ein Spiel vor, das nicht nur von reinem Text, sondern auch von varitativen Grafik- und Soundeffekten lebt. Die Hires Grafik konnte ich aufgrund Speicherplatzmangels nicht einsetzen, aber ich holte alles raus, was die Grafik des Textmodus hergab. „Rems" habe ich zu Gunsten des Programminhaltes nur die allernötigsten eingefügt. Auf Grund des Speicherplatzes im C 16 habe ich auch manche Zeilen (If-Abfragen) recht vollgepackt, doch auch wenn diese Zeilen über 80 Zeichen zu haben scheinen, es erscheint bei keiner ein „String too long-error", selbst wenn man sie ohne Befehlsabkürzungen mit Return integriert.

**I. Zum Spielverlauf:**
Sie sind in den Krater des Vulkans Baracuda gefallen und müssen sehen, daß Sie schnellstens wieder herauskommen, denn der Ausbruch steht kurz bevor. Der Krater hat viele Gänge und man weiß nie, welcher der jeweils kürzeste Weg ist. Es gibt aber einige Monster, die ihre Fallen in vielen Gängen ausgelegt haben und wenn Sie diesen Ungeheuern in die Finger geraten, wird es schwer werden, die Aufgaben zu bestehen, um nicht getötet zu werden. Haben Sie es jedoch geschafft, an einen Ausgang des Vulkans zu gelangen (vielleicht durch die Hilfe des smaragdgrünen Einhorns), erwartet Sie ein lautes und farbenfrohes Finale.

**II. Programmaufbau:**
Zuerst das Hauptprogramm. Dann die Spielunterprogramme: (Buchstabenraten), (Grausame Gabler), (Namenchanger), (Loserbuster), (Freßmonster), (Steinschlag), (Wilde Schweinchen) und (smaragdgrünes Einhorn). Die Unterprogramme zum Zeichnen des Vulkans, des Finales und die zahlreichen Unterunter- und Unterunterunterprogramme habe ich bei dieser Aufzählung nicht berücksichtigt.

**III. Detaillierte Erklärungen zum Programm:**
Durch den "TRAP"-Befehl habe ich eine Möglichkeit für diejenigen Leute eingebaut, die es leid sind, den Vorspann und die Einleitung zum xten Mal zu ertragen. Startet man das Programm mit "RUN" und drückt danach die "Run-Stop-Taste", springt der Computer sofort in Zeile 460 ohne Verlust der definierten Strings.
Der Pokebefehl setzt die Anzahl der im Tastaturpuffer vorhandenen Zeichen auf 0.
Wenn die in Zeile 560/570 erzeugte Zufallszahl 9 oder 10 beträgt, wird die Schleifenvariable, die die Zahl der zurückgelegten Schritte anzeigt, um 1 erhöht. Ergebnis: Sie kommen einen Schritt weiter, ohne daß irgendetwas passiert.
Wenn die Zufallszahl zwischen 1 und 8 beträgt, springt der Computer in das entsprechende (Spiel-)Unterprogramm.
Die Zuweisung ("String"=STR"String"(15-T)) wandelt die Zahl (15-T) in einen String um. Das ist notwendig, da die Anweisung "Char" die hier sehr praktisch ist, nur Text in Hochkommas oder eben Strings akzeptiert.
Der String FE $ taucht am Ende eines jeden Spieles auf. Hat man das Spiel bestanden (gewonnen), erhält er den Wert "R", hat man versagt den Wert "F". Im ersten Fall geht das Programm normal weiter, im zweiten jedoch wird der Computer zurückgeschickt und endet damit. Die Anweisungen in diesem Abschnitt bewirken, daß zweimal „gewürfelt" wird.
Hier wird das gemeine Lachen des grausamen Gamblers am Anfang und am Ende, falls man verloren hat, produziert.
Eine Zufallszahl wird zwischen 3072 und 4072 (Bildschirmspeicher) erzeugt. In den Zeilen danach wird ausgeschlossen, daß das spätere Wort, dessen Position durch "AD" bestimmt wird, zu nahe an den rechten Bildschirmrand gedruckt wird, so daß das Wort nicht zweigeteilt auftritt, außerdem ist es optisch auch schöner. Es wird ein zufälliges Buchstabenpokecode erzeugt, der dann an die vorher bestimmte (AD) Adresse gepokt wird. Der Name, den der Spieler für den richtigen hält, wird eingelesen und mit dem vorher zufällig bestimmten Namen verglichen. Dann springt der Computer in das jeweilige Unterprogramm zum Zeichnen einer Karte, das durch eine Zufallszahl bestimmt wurde. Übrigens stehen bei diesem Spiel praktisch zwei vollständige Herzsequenzen zur Verfügung, da ich eine einmal aufgetauchte Karte nicht von der Liste „streiche".
Es werden die beiden Anfangsbuchstaben der zu erratenden Karte eingelesen und mit den tatsächlichen verglichen und ausgewertet. Es werden die Variablen, die bekunden, ob die Zahl für das jeweilige Freßmonster gefunden wurde, auf 0 gesetzt und die Zufallszahlen für die Monster determiniert.
Es werden die jeweiligen Zeichen auf den Bildschirm gepoket. Es geht einfacher, doch um das Basic V 3.5 auszunutzen, arbeite ich hier mit "Trap": Es wird eine Zahl eingelesen, die zwischen 1 und 7 liegen muß. Durch "Trap" erscheint bei allen Fehlersituationen (z.B. "Run-Stop" oder einem "Type Mismatch") "Nur die Tasten von 1 – 7".
Es werden beim Bewegen der Monster die vorherigen Positionen gelöscht. Es werden durch die IF-Abfrage die Monster bewegt, oder wenn eine Zahl gefunden wurde, „gestrichen". Es werden die durch Zufallszahlen erhaltene Bildschirmpositionen durch das reverse Herz und "5"mal durch die Steine besetzt.
Nun wird die „Herzposition" mit den jeweiligen „Steinpositionen" verglichen. Bei Übereinstimmung sind Sie für den Computer gestorben.
Wenn Sie den Ausgang des Vulkans erreichen sollten, freut sich ein Männchen mit Ihnen und hüpft Hampelmannähnlich herum. Das Einhorn beschert Ihnen ein Weiterkommen von einem bis zu acht Schritten, während es unter dem Text vorbeigaloppiert.
Manche Zeilen enthalten Subroutinen, die so oft vorkommen, daß ich sie als Unterprogramme integriert habe.
Es werden der Vulkan gezeichnet und durch Zufall Zeichen erzeugt, die der Vulkan zufällig „speit".

**IV: Liste von Basic-Befehlen, die vielleicht nicht jedem geläufig sind:**
TAB(X): Ein Text wird um X Positionen (0–255) verschoben gedruckt.
GETKEY X "String": Das Programm wartet, bis eine Taste gedrückt wird (entspricht beim C-64 WAIT 198,1).
IF...THEN...:ELSE: IF-Abfrage mit Alternative.
COLOR 1,X,Y: Die Zeichenfarbe (oder: 0-Hintergrund; 4=Rahmen) wird auf die Farbe X mit der Helligkeitsstufe Y gesetzt.
SOUND X,Y,Z: Der Ton wird mit der Dauer Z durch den Tongenerator X angesprochen.
VOL(X): Ein Ton wird mit der Lautstärke X gespielt.

*Bitte lesen Sie weiter auf Seite 42*

```
10 rem baracuda =============plus 4
20 rem (p) commodore welt       ts=
30 rem ============================
40 rem (c) by                     =
50 rem martin purucker            =
60 rem wuerzburg                  =
70 rem version 3.5  40z/ascii     =
80 rem plus 4 oder c 16 + 64 k    =
90 rem ============================
100 r$=chr$(018):o$=chr$(146):f$=ch
r$(130):g$=chr$(131)
110 es$=chr$(027):ho$=chr$(019):z0$
=chr$(177)
120 trap310:gosub1390
130 fort=1to1000:next:color 1,1
140 char1,39,22,es$+"b":char1,0,13,
es$+"t":rem windowing *
150 scnclr
160 print"sie armer teufel sind in
den krater ge-":print
170 print"fallen und muessen schnel
lstens hier ":print
180 gosub4380
190 print"raus,denn der ausbruch st
eht kurz bevor!":print
200 gosub4370
210 print"sie koennen nach sueden,n
orden,westen":print
220 print"oder osten mit s,n,w,o fl
uechten.":print
230 gosub4380:scnclr
240 print"aber vorsicht ! - es gibt
 viele monster":print
250 print"und fallen in den gaengen
 des vulkans !":print
260 gosub4380
270 print"jede konfrontation gefaeh
rdet ihr leben!":print
280 gosub4380
290 char1,9,20,f$+"viel glueck ! !
!"+g$
300 gosub4370
310 printho$ho$:color 1,1
320 xx=int(rnd(1)*16)+5
330 scnclr
340 fors2=1toxx
350 aa=xx-s2+1:poke239,0
360 ifaa=0thenxx=s2-1:goto560
370 print:print:print"sie muessen g
enau";aa;"schritte bis zum":print
380 print"ausgang gehen !":gosub422
0
390 scnclr:print:print"in welche ri
chtung wollen sie gehen ?...(s/n/w/
o)"
400 getkey ri$
410 ifri$="s"orri$="o"thenri=int(rn
d(0)*10)+1:goto440
420 ifri$="n"orri$="w"thenri=int(rn
d(1)*10)+1:goto440
430 print:print"depp !":fort=1to100
0:next:goto390
440 ifri=9orri=10thens2=s2+1:goto35
0
450 ifs2=xxthen570
460 onrigosub620,810,1490,1920,3310
,3660,3940,4090:color1,1
470 ifri=7then340
480 iffe$="f"thenxx=s2-1:goto550
490 scnclr
500 print:printtab(8)f$;"gratuliere
 !";g$
510 print:print"sie haben das abent
euer bestanden ! -":print
520 print"es geht weiter !"
530 nexts2
540 goto560
550 print:print:print"tut mir leid,
sie werden eingeaeschert !":end
560 scnclr
570 print:printf$;"super - spitze -
 klasse";g$:print
580 gosub4370
590 print:print"sie haben es gescha
fft ! ! !":print
600 print"ihr leben ist gerettet !
! !":gosub4380:scnclr:gosub3990
610 end
620 rem ****** buchstabenraten *
630 scnclr:print:print"ihr leben is
t in gefahr ! - "
640 print:print"ihre einzige chance
 zu ueberleben ist,":print
650 print"wenn sie aus den 26 buchs
taben in 15 "
660 print:print"versuchen den richt
igen finden ! ! !":print
670 printf$;"viel glueck";g$
680 b$=chr$(int(rnd(1)*26)+65)
690 fort=1to15
700 getkey a$:vol8:sound1,90,5:ifa$
=b$thenx=t-1:goto760
710 c$=str$(15-t):char
1,5,12,"noch"+c$+" versuche ":next:
print
720 sound1,0,100:sound2,500,100
730 print:print:print"tut mir leid,
sie sind tot !":gosub4370
740 fe$="f"
750 return
760 print:print:printtab(10)f$;"guu
ut !";g$
770 fort=0to500:sound1,t,1:sound2,t
+500,1:next
780 gosub4220
790 fe$="r"
```

```
800 return
810 rem ***** grausamer gambler *
820 scnclr:print:print"ich bin der
grausame gambler,und du":print
830 print"bist in meine folterhoehl
e geraten ! ! !":print
840 print"du kommst hier nur lebend
 wieder heraus":print
850 print"wenn du mich im wuerfelra
tespiel besie-":print
860 print"gst. also dann wollen wir
 mal um dein":print
870 printtab( 15)f$;"leben";g$:print
:print"spielen !"
880 gosub4370:gosub1310
890 gosub4220
900 scnclr
910 forx=1to2
920 fori=1to6
930 h( i)=0
940 nexti
950 fori=1to2
960 w=int( 6*rnd( 1))+1
970 h( i)=w
980 nexti
990 print h( 1);"+";h( 2);"=";h( 1)+h(
2)
1000 f( x)=h( 1)+h( 2)
1010 gosub4370
1020 print
1030 print
1040 ifx=2then1100
1050 print"raten sie : wird ihr wur
f hoeher, oder":print
1060 print"niedriger werden ? (h/n)
"
1070 getkey w$
1080 print
1090 nextx
1100 iff( 1)-f( 2)>0andw$="h"orf( 1)-f
( 2)<0andw$="n"thenprint:print:goto1
140
1110 iff( 1)-f( 2)=0thenprint"oh,eins
tand,naja ein neues spiel-ein neues
 leben !":goto1130
1120 printtab( 13)f$;"gewonnen !";g$
:goto1170
1130 gosub4370:goto900
1140 print:print:printtab( 12)f$;"ve
rloren !";g$:gosub1310
1150 fe$="f"
1160 return
1170 vol5
1180 sound1,4,15
1190 sound1,60,14
1200 sound1,150,13
1210 sound1,320,12
1220 sound1,550,15
1230 sound1,550,15
1240 sound1,320,12
1250 sound1,150,13
1260 sound1,60,14
1270 sound1,4,15
1280 sound1,0,20
1290 fe$="r"
1300 return
1310 print:print:print
1320 printtab( 10)f$;r$;"ha, ha, ha
! ! !";o$;g$
1330 forb=1to3
1340 forh=1to1000step20
1350 vol5
1360 sound1,h,1
1370 nexth
1380 forj=1to300:nextj:nextb:print:
return
1390 rem *********** vorspann *
1400 color1,3,4:scnclr:print:gosub4
240
1410 print
1420 volint( rnd( 1)*8)+1
1430 d=int( rnd( 1)*16)+1
1440 color1,d,int( d/3)
1450 printtab( 10)f$"das ist baracud
a,";g$
1460 print
1470 printtab( 3)f$;"der allesvernic
htende vulkan ! ! !";g$
1480 fort=1to10:sound1,4,2:sound3,0
,20:nextt:return
1490 rem ******** namenchanger *
1500 scnclr:color1,1
1510 print:print"du bist hier im lo
ch von namenchanger,":print:print"u
nd ich gebe dir";
1520 print"eine chance,":print:prin
t"dein erbaermliches leben zu rette
n ! ! !"
1530 print:print
1540 print"irgendwo auf diesem bild
schirm wird":print
1550 print"gleich mein 4-stelliger
name kurz auf-":print
1560 print"leuchten ! - merke..ihn
dir !!!!!!!!!!!":print
1570 print
1580 gosub4220
1590 scnclr
1600 ad=int( rnd( 1)*1000)+3072
1610 forp=1to25
1620 ifad=3071+p*40thenad=ad-9
1630 ifad=3070+p*40thenad=ad-8
1640 ifad=3069+p*40thenad=ad-7
1650 ifad=3068+p*40thenad=ad-6
1660 nextp
1670 fors=1to4
```

```
1680 ad=ad+1
1690 ze=int(rnd(1)*26)+1
1700 a(s)=ze
1710 pokead,ze
1720 nexts
1730 fort=1to250:next:scnclr
1740 print:print"wie ist denn nun m
ein name ?"
1750 print:print:print:print
1760 getkey n1$,n2$,n3$,n4$
1770 name$=n1$+n2$+n3$+n4$
1780 a$=chr$(a(1)+64):b$=chr$(a(2)+
64):c$=chr$(a(3)+64):d$=chr$(a(4)+6
4)
1790 geamt$=a$+b$+c$+d$
1800 ifname$=gesamt$then1880
1810 print"ha, ha - falsch - zuerst
 werfe ich dich":print
1820 print"den wilden teufeln vor,
und dann kommst":print
1830 print"kommst du in den ";f$;"k
ochtopf !";g$
1840 print
1850 printtab(11)f$;r$"har, har, ha
r !";g$;o$:gosub4370
1860 fe$="f"
1870 return
1880 print"grxymkpfl - richtig - du
 bist frei und":print
1890 print"kannst weitergehen !":go
sub4220
1900 fe$="r"
1910 return
1920 rem ********* loserbuster *
1930 scnclr
1940 print:print"darf ich mich vors
tellen:":print
1950 print"ich bin loserbuster,und
ich hasse ver-":print
1960 print"lierer so, dass ich sie
in streifen":print
1970 print"schneide und in die spei
sekammer haenge!"
1980 print:print
1990 print"wie waere es denn mit ei
nem spielchen ?!":print:print
2000 print"ah, ich sehe, sie wollen
 mitspielen -":print
2010 print"also gut !":gosub4220
2020 scnclr
2030 print"das ist ihre karte : "
2040 print:for t=1to1000:next
2050 ka=int(rnd(0)*9)+1
2060 onkagosub2320,2430,2540,2650,2
760,2870,2980,3090,3200
2070 print:gosub4370
2080 color1,1:print"na, sind sie mi
t ihrer karte zufrieden ?":print
2090 gosub4370:scnclr
2100 print:print"auch wenn ich jetz
t eine hoehere karte":print
2110 print"habe,koennen sie immer n
och";
2120 printf$;" gewinnen !";g$:print
2130 print
2140 print"sie muessen nur erraten,
 welche karte":print:print"ich habe
":print
2150 print:print"geben sie die zwei
 anfangsbuchstaben der":print
2160 print"karte ein (6=se bis as=a
s)"
2170 getkey z1$,z2$
2180 k2=int(rnd(0)*9)+1:print:print
"das ist meine karte :":print
2190 onk2gosub2320,2430,2540,2650,2
760,2870,2980,3090,3200:color1,1
2200 if(ka-k2)>0then2270
2210 ifz1$=z3$andz2$=z4$then2270:pr
int
2220 print
2230 if(ka-k2)=0thenprint"im zweife
l fuer mich -":print:print"tja, pec
h gehabt!"
2240 gosub4370:char1,15,18,f$+"v e
r l o r e n ! ! !"+g$:gosub4370
2250 fe$="f"
2260 print:print:print:print:print:
return
2270 print"gratuliere, sie haben ge
wonnen, und":print:print"koennen un
behelligt gehen !"
2280 gosub4220
2290 fe$="r"
2300 return
2310 rem ******* kartengraphic *
2320 color 1,3,3
2330 printtab(5)"UCCCCCI"
2340 printtab(5)"B6    B"
2350 printtab(5)"B S S B"
2360 printtab(5)"B     B"
2370 printtab(5)"B S S B"
2380 printtab(5)"B     B"
2390 printtab(5)"B S S B"
2400 printtab(5)"B    6B"
2410 printtab(5)"JCCCCCK"
2420 z3$="s":z4$="e":return
2430 color1,3,3
2440 printtab(5)"UCCCCCI"
2450 printtab(5)"B7    B"
2460 printtab(5)"B S S B"
2470 printtab(5)"B     B"
2480 printtab(5)"B S S B"
2490 printtab(5)"B     B"
2500 printtab(5)"B S S B"
2510 printtab(5)"B    7B"
```

```
2520 printtab(5)"JCCCCK"
2530 z3$="s":z4$="i":return
2540 color1,3,3
2550 printtab(5)"UCCCCI"
2560 printtab(5)"B8    B"
2570 printtab(5)"B S S B"
2580 printtab(5)"B     B"
2590 printtab(5)"B S S B"
2600 printtab(5)"B     B"
2610 printtab(5)"B S S B"
2620 printtab(5)"B    8B"
2630 printtab(5)"JCCCCK"
2640 z3$="a":z4$="c":return
2650 color1,3,3
2660 printtab(5)"UCCCCI"
2670 printtab(5)"B9    B"
2680 printtab(5)"B S S B"
2690 printtab(5)"B     B"
2700 printtab(5)"B S S B"
2710 printtab(5)"B     B"
2720 printtab(5)"B S S B"
2730 printtab(5)"B    9B"
2740 printtab(5)"JCCCCK"
2750 z3$="n":z4$="e":return
2760 color1,3,3
2770 printtab(5)"UCCCCI"
2780 printtab(5)"B10   B"
2790 printtab(5)"B S S B"
2800 printtab(5)"B     B"
2810 printtab(5)"B S S B"
2820 printtab(5)"B     B"
2830 printtab(5)"B S S B"
2840 printtab(5)"B   10B"
2850 printtab(5)"JCCCCK"
2860 z3$="z":z4$="e":return
2870 color1,3,3
2880 printtab(5)"UCCCCI"
2890 printtab(5)"Bb    B"
2900 printtab(5)"B S S B"
2910 printtab(5)"B     B"
2920 printtab(5)"B S S B"
2930 printtab(5)"B     B"
2940 printtab(5)"B S S B"
2950 printtab(5)"B    bB"
2960 printtab(5)"JCCCCK"
2970 z3$="b":z4$="u":return
2980 color1,3,3
2990 printtab(5)"UCCCCI"
3000 printtab(5)"Bd    B"
3010 printtab(5)"B S S B"
3020 printtab(5)"B     B"
3030 printtab(5)"B S S B"
3040 printtab(5)"B     B"
3050 printtab(5)"B S S B"
3060 printtab(5)"B    dB"
3070 printtab(5)"JCCCCK"
3080 z3$="d":z4$="a":return
3090 color1,3,3
3100 printtab(5)"UCCCCI"
3110 printtab(5)"Bk    B"
3120 printtab(5)"B S S B"
3130 printtab(5)"B     B"
3140 printtab(5)"B S S B"
3150 printtab(5)"B     B"
3160 printtab(5)"B S S B"
3170 printtab(5)"B    kB"
3180 printtab(5)"JCCCCK"
3190 z3$="k":z4$="o":return
3200 color1,3,3
3210 printtab(5)"UCCCCI"
3220 printtab(5)"Ba    B"
3230 printtab(5)"B S S B"
3240 printtab(5)"B     B"
3250 printtab(5)"B S S B"
3260 printtab(5)"B     B"
3270 printtab(5)"B S S B"
3280 printtab(5)"B    aB"
3290 printtab(5)"JCCCCK"
3300 z3$="a":z4$="s":return
3310 rem ******* fressmonster *
3320 scnclr:print
3330 print"vorsicht - sie sind in d
ie hoehle der":print
3340 printf$;"fressmonster ";g$;"ge
raten !":print
3350 print"sie koennen den monstern
 nur entkommen,":print
3360 print"wenn sie aus den zahlen
1-7 in fuenf ver":print
3370 print"suchen die richtigen zah
len finden,um":print:print"sie zu s
toppen ! ! !"
3380 print
3390 print"bei jedem tastendruck ko
mmen sie naeher!":char1,10,18,f$+"g
ood luck !"+g$
3400 print:gosub4220
3410 e1=0:e2=0:e3=0:e4=0:forv=1to4:
o(v)=int(rnd(1)*7)+1:next
3420 scnclr
3430 poke3171,0:poke3371,211:poke33
76,0:poke3366,0:poke3571,0
3440 fort=1to5
3450 ife1=1ande2=1ande3=1ande4=1the
n3560
3460 trap3590:getkey az
3470 ifaz<1oraz>7then3590
3480 gosub3600
3490 vol5:sound3,0,10
3500 nextt
3510 ife1=1ande2=1ande3=1ande4=1the
n3560
3520 vol8:sound1,900,40:foryy=1to3:
forc=1to500:next:sound3,0,5:nextyy
3530 char1,1,12,"pech gehabt !":gos
ub4370
```

```
3540 fe$="f"
3550 return
3560 char1,1,12,"noch mal dem tod e
ntronnen !":print:gosub4220
3570 fe$="r"
3580 return
3590 char1,1,14,"nur die tasten von
 1-7 !":gosub4370:goto3410
3600 poke3171+(t-1)*40,96:poke3376-
(t-1),96:poke3366+(t-1),96:poke3571
-(t-1)*40,96
3610 ifaz<>o(1)ande1=0thenpoke3171+
t*40,0:elsee1=1
3620 ifaz<>o(2)ande2=0thenpoke3376-
t,0:elsee2=1
3630 ifaz<>o(3)ande3=0thenpoke3366+
t,0:elsee3=1
3640 ifaz<>o(4)ande4=0thenpoke3571-
t*40,0:elsee4=1
3650 return
3660 rem ********* steinschlag *
3670 scnclr:print:print"achtung!-di
e ersten auswirkungen des be-":prin
t
3680 print"vorstehenden ausbruches
sind im anrollen":print:print" - st
einschlag ! ! ! "
3690 print:print:print"sie sind das
 revers dargestellte S und":print
3700 print"koennen nur hoffen,dass
sie von den ":print
3710 print"steinen verschont bleibe
n ! - "
3720 print:print
3730 printtab(3)f$;"v i e l  g l u
e c k ! ! !";g$:print:print:gosub42
20
3740 scnclr:color1,3,4
3750 a=int(rnd(1)*1000)+3072
3760 pokea,83+128
3770 fort=1to1000:next
3780 s=int(rnd(1)*500)+300
3790 color1,1:forx=1tosstep1
3800 volint(rnd(1)*4)+1:sound3,int(
rnd(1)*10),1
3810 p=int(rnd(1)*1000)+3072
3820 pokep,81
3830 ifp=athens=x-1:vol8:sound1,0,5
0:goto3890
3840 nextx
3850 gosub4370
3860 scnclr:print:printtab(6)f$;"gl
ueck gehabt..! ! !";g$:gosub4220
3870 fe$="r"
3880 return
3890 gosub4370:print:scnclr
3900 print:print"das war pech - sie
sind":print:printtab(9)f$;"zermant
scht ! ! !";g$
3910 gosub4370
3920 fe$="f"
3930 return
3940 rem ********** schweinchen *
3950 print:print"sie werden von wil
den schweinchen ver-":print
3960 print"folgt und muessen in den
 krater zurueck":print:printtab(15)
"! ! !"
3970 gosub4380
3980 s2=0:return
3990 rem ************* finale *
4000 gosub4240:color1,12,4:fors=1to
10
4010 char1,14,10,"MWN":char1,14,11,
" ":char1,14,12,"N M"
020 for t=1to10:vol8:sound1,10*t+9
0,1:next
030 char1,14,10," W ":char1,14,11,
 ":char1,14,12,"    "
040 fort=1to100:next:next
050 vol8:formm=1to20:sound1,int(rn
(1)*300),int(rnd(1)*20)
060 color0,int(rnd(1)*16)+1,int(rn
(1)*3)+3:sound1,int(rnd(1)*300)+20
,8:next
070 gosub4370:vol0
080 scnclr:color1,1:color0,2:retur

090 rem **************** einhorn *
100 sw=int(rnd(1)*7)+1:print:print
color1,6,4
110 print"ein smaragdgruenes einho
n gallopiert":print
120 print"an ihnen vorbei ! sie ko
nnen aufsprin-"
130 print:print"gen und kommen";sw
1;"schritte weiter !"
140 print:print:print:print" ";:fo
t=1to250:next
150 forbb=1to19
160 color1,2:printchr$(157);" ";:c
lor1,6,4:printchr$(29);" ";
170 for t=1to200:next
180 nextbb:color1,2:printchr$(157)
" "
190 color1,1:gosub4220
200 s2=s2+sw
210 return
220 print:print"wenn sie bereit si
d, ";r$;"space";o$:getkey
t$:return
230 rem ******** vulkangraphic *
240 print:print:print:print:print
250 printtab(6)"...N.........M"
4260 printtab(6)"..N...........M"
4270 printtab(6)".N.............M"
```

```
4280 printtab(6)"N......J   K,.....
M"
4290 printtab(5)"N.................
..M"
4300 printtab(4)"N.................
....M"
4310 forq3=1to50
4320 vol8:sound1,int(rnd(1)*200)+80
0,1:sound3,0,5
4330 q2=int(rnd(1)*8):q0=int(rnd(1)
*14)+3080
4340 pokeq0+40*q2,int(rnd(1)*44)+84
4350 nextq3
4360 return
4370 fort=1to2000:next:return
4380 fort=1to4000:next:return
4390 rem baracuda ============plus 4
4410 rem 011797 bytes program    ===
4420 rem 000000 bytes variables ===
4430 rem 000000 bytes arrays     ===
4440 rem 000000 bytes strings    ===
4470 rem ========================
```

# BARACUDA

TRAP X: Wenn ein Fehler (TYPE MISMATCH, RUN-STOP, ...) auftreten sollte, springt das Programm zu Zeile X und steigt nicht aus.
SCNCLR: Der Bildschirm (ob Text- oder Graphicmodus) wird gelöscht.
ON X GOTO (GOSUB) Y, Z, ...: Wenn X=1, dann springt das Programm zu Zeile Y, wenn X=2, zu Zeile Z, usw.
CHAR X,Y,Z,Q: Der Text (String) Q wird in Spalte Y und Zeile Z in der Farbe gedruckt, auf die die Farbzonennummer X gesetzt ist.

**V: Liste der Sonderstrings:**

| | | | |
|---|---|---|---|
| R "String"=CHR "String" | (18) | = | Revers an |
| O "String"=CHR "String" | (146) | = | Revers aus |
| F "String"=CHR "String" | (130) | = | Flash an |
| G "String"=CHR "String" | (131) | = | Flash aus |
| G "String"= CHR"String" | (27) | = | Escape-Taste |
| E "String"=CHR "String" | (19) | = | Home-Taste |
| CHR "String" | (29) | = | Cursor rechts |
| CHR "String" | (157) | = | Cursor links |

*Martin Purucker*



# DIAGRAMME

Auch mit den Kleinen aus der Commodore-Familie kann man statistische Daten (z.B. Wahlergebnisse) auf fast schon professionelle Weise grafisch darstellen, wie es das Balkendiagramm für den C 16 und (Aufwärts-) kombatible Computer zeigt.

Selbstverständlich kann man die nötigen Werte auch von einem externen Datenträger einlesen, sofern sie in entsprechender Form aufgezeichnet sind.

Die Säulen werden in Kavalierperspektive so auf den Bildschirm gebracht, daß dieser optimal ausgenutzt wird ohne die Übersichtlichkeit der Darstellung zu beinträchtigen. Über die einzelnen Säulen wird der zugehörige Wert und darunter die Beschriftung in das Diagramm eingetragen. Auch eine zentrierte Über- und Unterschrift ist möglich.

**Variablenliste**
N, W(I), F(I), BS(I); MAX L - L = Maximalwert; T, ZX, ZY – Hilfsgrößen zur Bildschirmeinteilung;
ZW - Zwischenraum zw. 2 Säulen; DX - Breite der Säulen; DH, H - Einheit, bzw. Höhe der Säule; Y - Untere Säulenbegrenzung (Y-Richtung); AX - Aktueller X-Wert; SP, ZE, L - Spalte, Zeile, Länge

# TELEFON-DATEI

Diese Telefondatei ermöglicht dem C-16-Benutzer endlich die Verwaltung von Telefonnummern und die Überprüfung der Telefonkosten in einem Programm! Nach Start des Programmes wird im Menü ausgewählt. Hierbei ist erstens die Neueingabe von Telefonnummern (natürlich mit dem dazugehörigen Namen) möglich, zweitens das Durchblättern der bereits gespeicherten Daten und drittens die Gebührenerfassung.
Zu 1: Nach der vollständigen Eingabe einer Adresse schreibt das Programm selbsttätig diese Adresse in eine Data-Zeile und hängt diese an das Programm an. Daher nach jeder Benützung das Programm neu absaven, bevor Sie es aus dem Rechner "rausschmeißen", um die Daten nicht zu verlieren.
Zu 2: Mit einer beliebigen Taste kann hier weitergeblättert werden. um die Datei durchlaufen zu lassen.
Zu 3: Dieser Menüpunkt unterscheidet "Teledat" von den herkömmlichen Datei-Programmen. Der Benützer wählt zuerst den entsprechenden Wochentag sowie die Uhrzeit aus (damit Tele-Dat zwischen den verschiedenen Tarifen unterscheiden kann) und drückt dann bei Gesprächsbeginn eine Taste. Im oberen Bildschirmteil erscheint nun eine laufende Stoppuhr, darunter werden die bereits „verbrauchten" Einheiten sowie die Gesamtsumme angezeigt. Bei Ende des Telefonates muß wieder eine beliebige Taste gedrückt werden, damit die Berechnung beendet wird. Nun können die Gesamtsumme, Gesprächsdauer und die Gebühreneinheiten abgelesen werden.

```
10 rem balkendiagramm =========v3.5
20 rem (p) 05/85 cbm revue team  =
30 rem ===========================
40 rem (c) 04/85 by paul saar    =
50 rem             muehlheim/ruhr =
60 rem                            =
70 rem cbm-version 3.5            =
80 rem c-116 + 1531/1541          =
90 rem ===========================
100 re$=chr$(28):bl$=chr$(31)
110 gr$=chr$(30):c4$=chr$(17)
120 color 0,2
130 graphic 0,1
140 color 1,7
150 color 4,2
160 printre$"........balkendiagramm
"
170 print""
180 printgr$" mit 10 darstellungsmo
eglichkeiten"
190 for t=1 to 4000:next t
200 print""
210 printbl$".angaben.der.werte.in.
%"
220 print""
230 for t=1 to 4000:next t
240 vol 8
250 sound 1,866,500
260 printre$" keine werte ueber 100
% eingeben"
270 for t=1 to 6000:next t
280 printre$
290 scnclr
300 color 1,1
310 input" wert a in %   ";x1
320 input" wert b in %   ";x2
330 input" wert c in %   ";x3
340 input" wert d in %   ";x4
350 input" wert e in %   ";x5
360 input" wert f in %   ";x6
370 input" wert g in %   ";x7
380 input" wert h in %   ";x8
390 input" wert i in %   ";x9
400 input" wert j in %   ";x0
410 graphic 2,1
420 color 1,1
430 draw 1,40,50to40,155,35,150to30
0,150,35,50to300,50,35,60to300,60,3
5,70to300,70
440 draw 1,35,80to300,80,35,90to300
,90,35,100to300,100,35,110to300,110
,35,120to300,120
450 draw 1,35,130to300,130,35,140to
300,140
460 char 1,1 , 6,"100"
470 char 1,1,18,"  0"
480 char 1,1,12," 50"
490 char 1,5,1,"balkendiagramm"
500 char 1,5,3,"prozentwerte spalte
 a-j"
510 a=150-x1
520 color 1,7,1
530 box 1,48,a,64,150,,1
540 b=150-x2
550 color 1,3,4
560 box 1,72,b,88,150,,1
570 c=150-x3
580 color 1,6,4
590 box 1,96,c,112,150,,1
600 d=150-x4
610 color 1,8,2
620 box 1,120,d,136,150,,1
630 e=150-x5
640 color 1,10,4
650 box 1,144,e,160,150,,1
660 f=150-x6
670 color 1,8,6
680 box 1,168,f,184,150,,1
690 g=150-x7
700 color 1,16,2
710 box 1,192,g,208,150,,1
720 h=150-x8
730 color 1,12,2
740 box 1,216,h,232,150,,1
750 i=150-x9
760 color 1,4,4
770 box 1,240,i,256,150,,1
780 j=150-x0
790 color 1,1
800 box 1,264,j,280,150,,1
810 char 1,6,19,"a  b  c  d  e  f
g  h  i  j"
820 printc4$re$"noch einmal ?  (j/n
)"
830 geta$:ifa$="j"then100
840 ifa$="n"thenend
850 goto830
860 rem balkendiagramme =======v3.5
870 rem 60671 bytes memory    =====
880 rem 02138 bytes program   =====
890 rem 00000 bytes variables =====
900 rem 00000 bytes arrays    =====
910 rem 00000 bytes strings   =====
920 rem 58533 bytes free      =====
```

```
10 rem teledat===================p4
20 rem (c)        cbm revue team   =
30 rem =============================
40 rem (c)        by               =
50 rem                             =
60 rem bernd welte                 =
70 rem                             =
80 rem gebuehrenerfassung von      =
90 rem herrman wellesen            =
100 rem                            =
110 rem c16/116/p4 + floppy/cass.  =
120 color4,7:color0,7
130 rn$=chr$(18):he$=chr$(19)
140 rf$=chr$(146):cl$=chr$(147)
150 c4$=chr$(17):s1$=chr$(32)
160 z1$=chr$(42)
170 forq=1to3:q3$=q3$+c4$:nextq
180 forq=1to4:q4$=q4$+c4$:nextq
190 forq=1to6:q6$=q6$+c4$:nextq
200 forq=1to7:q7$=q7$+c4$:nextq
210 forq=1to8:q8$=q8$+s1$:nextq
220 forq=1to40:qa$=qa$+z1$:nextq
230 forq=1to40:qb$=qb$+s1$:nextq
240 forq=1to38:qc$=qc$+s1$:nextq
250 printcl$;
260 printqa$;
270 printz1$qc$z1$;
280 printz1$tab(13)"a&b welte softw
are"q8$z1$;
290 printz1$qc$z1$;
300 printqa$;
310 printqb$;
320 printtab(15)"***********"
330 printtab(15)"*tele-dat*"
340 printtab(15)"***********"
350 printqb$;
360 printqb$;
370 printtab(11)"(1) nummerneingabe
"
380 printqb$;
390 printtab(11)"(2) gebuehrenerfas
sung"
400 printqb$;
410 printtab(11)"(3) ausgabe"
420 printqb$;
430 printtab(11)"(4) ende"
440 printqb$;
450 printqb$;
460 printqb$;
470 printtab(17)"(c) 1985"
480 print
490 do
500 getkeya$
510 ifa$="1"then560
520 ifa$="2"then930
530 ifa$="3"then810
540 ifa$="4"thenend
550 loop
560 printchr$(147):printtab(9)"frei
e bytes "chr$(18);fre(x):print:prin
t:printchr$(18)tab(15)"0= menue"
570 print:print:print"...vor-und na
chname":print:inputv$:ifv$="0"then1
0
580 print:print"..vorwahlnr.":print
:inputvn$
590 print:print"..nummer":print:inp
utn$
600 char1,11,20,"alles richtig? (j/
n)........."
610 do
620 getkeyq$
630 ifq$="n"then560
640 ifq$="j"then670
650 loop
660 rem dataerzeugung
670 dt= 20001
680 printchr$(147);dt;"data";v$;","
;vn$;",";n$
69A dt=dt+1
700 print"670 dt=";dt
710 print"goto 750"
720 poke1319,19
730 fori=1to4:poke1320+i,13:next
740 poke239,4:stop
750 printchr$(147):char1,9,20,"weit
er eingeben ? (j/n)"
760 do
770 getkeyq$
780 ifq$="j"then560
790 ifq$="n"thenrun
800 loop
810 restore
820 readv$,vn$,n$
830 ifv$="00"thenrun
840 gosub860
850 goto820
860 printcl$q6$s1$;v$
870 printc4$c4$" tel. ";vn$;" / ";n
$
880 char1,0,20,"...........weiter b
itte taste........."
890 :
900 getkeyq$
910 :
920 return
930 rem gebuehrenerfassung
940 printcl$q3$"...waehlen sie..die
 entfernungszone:....";
950 printc4$s1$rn$" 0 "rf$s1$s1$"or
ts- und nahgespraeche"
960 printc4$s1$rn$" 1 "rf$s1$s1$"en
tfernung bis    50 km"
970 printc4$s1$rn$" 2 "rf$s1$s1$"en
tfernung bis...100 km"
980 printc4$s1$rn$" 3 "rf$s1$s1$"en
```

```
tfernung ueber 100 km"c4$c4$
990 inputz:z=int(z):ifz>3thenprintc
2$c2$:goto990
1000 printcl$q3$".....die uhrzeit:.
...."
1010 printc4$s1$rn$" 1 "rf$"..8 uhr
 bis 18 uhr"
1020 printc4$s1$rn$" 2 "rf$" 18 uhr
 bis  8 uhr"c4$c4$
1030 inputu:u=int(u):ifu>2oru<1then
printc2$c2$:goto1030
1040 printcl$q3$s1$"....den wochent
ag:...."
1050 printc4$s1$rn$" 1 "rf$" montag
 bis freitag"
1060 printc4$s1$rn$" 2 "rf$" samsta
g,sonntag und feiertage"c4$c4$
1070 inputd:d=int(d):ifd>2ord<1then
printc2$c2$:goto1070
1080 n=480:g=720
1090 ifz=1thenn=45:g=67.5
1100 ifz=2thenn=20:g=38.571
1110 ifz=3thenn=12:g=38.571
1120 e=n
1130 ifd=2oru=2thene=g
1140 printcl$;
1150 printq7$rn$" bei gespraechsbeg
inn taste druecken !.."; rf$
1160 getkeya$
1170 ti$="000000"
1180 printcl$s1$"dauer:"
1190 printc4$s1$s1$"...........gebu
ehreneinheiten"
1200 printq3$s1$s1$"...........gesp
raechskosten "
1210 t=val(left$(ti$,2))*3600+val(m
id$(ti$,3,2))*60+val(right$(ti$,2))
1220 b=int(t/e+1)
1230 b1=b*.23
1240 printhe$tab(8)left$(ti$,2)"."m
id$(ti$,3,2)"."right$(ti$,2)
1250 printc4$s1$s1$int(b)
1260 printq3$s1$s1$int(b1*100+.5)/1
00" dm "
1270 b=0:b1=0
1280 printrn$;
1290 char1,0,21,"..bei gespraechsen
de..taste druecken...."
1300 getz$:ifz$=""then1210
1310 char1,0,21,"...........noch ei
nmal? (j/n)..........."
1320 printrf$:do:getkeyz$
1330 ifz$="n"then250
1340 ifz$="j"then930
1350 loop
1360 rem===telefondatei=c16/116/p4
1370 rem 60671 bytes memory     ===
1380 rem 03551 bytes programm   ===
1390 rem 00119 bytes variables  ===
1400 rem 00000 bytes arrays     ===
1410 rem 02798 bytes strings    ===
1420 rem 56532 bytes fre        ===
1430 rem ==========================
50000 data00,00,00
```




**☏ TELEFONSERVICE ☏**

**Alle Experten der CBM REVUE/COMMODORE-WELT stehen unseren Lesern jeden Mittwoch zwischen 16.30 und 19.30 Uhr zur Beantwortung aller Fragen unter der Telefonnummer 089/1298013 zur Verfügung. Ebenso der Abo- und Kassettenservice. Einfach anrufen! 089/1298014!**


# WAS KOSTET IHR AUTO WIRKLICH?

Das Programm erklärt sich zum Teil von selbst. Der Benutzer kann das Programm an einem von ihm bestimmten Tag nur beginnen, wenn er sein Kfz an diesem Tag vollgetankt hat! Wenn das Kfz an diesem Tag nicht voll getankt ist, stimmen nachher einige Daten nicht mehr!
Die eingegebenen Daten müssen als sequentielle Dateien gespeichert werden, um später wieder mit den neuen Daten im Programm verarbeitet zu werden.

In den entsprechenden Zeilen müssen Sie nur noch die Daten Ihres Wagens eingeben. Kennzeichen, Typ usw. Die genannten Daten sind nur Beispiele und sind jederzeit veränderbar. »

```
10 rem supersound ===============p4
20 rem (p)        cbm revue team    =
30 rem ============================
40 rem (c)          by              =
50 rem jochen lellesch              =
60 rem                              =
70 rem version 3.5                  =
80 rem plus/4 + 1531/1541           =
90 rem ============================

100 z1$=chr$(96)
110 d1$=chr$(109):d2$=chr$(110)
120 forq=1to40:q1$=q1$+z1$:nextq
130 forq=1to20:q2$=q2$+d2$+d1$:next
q
140 forq=1to20:q3$=q3$+d1$+d2$:next
q
1000 scnclr:vol5
1010 sound1,917,75:sound2,810,75
1020 printq2$;
1030 printd1$"*********************
*****************"d2$;
1040 printd2$"****..................
.............****"d1$;
1050 printd1$"***...................
..............***"d2$;
1060 printd2$"**......s u p e r s o
 u n d...16....**"d1$;
1070 printd1$"***...................
..............***"d2$;
1080 printd2$"****..................
.............****"d1$;
1090 printd1$"*********************
*****************"d2$;
1100 sound1,810,75:sound2,596,75
1110 printd2$"......................
................."d1$;
1120 printd1$"*........ein musikpro
gramm von........"d2$;
1130 printd2$"**....................
...............**"d1$;
1140 printd1$"***...................
..............***"d2$;
1150 printd2$"**..j o c h e n....l
e l l e s c h **"d1$;
1160 printd1$"*.....................
................*"d2$;
1170 printd2$"......................
................."d1$;
1180
printd1$"*************************
************"d2$;
1190 sound1,596,75:sound2,169,75
1200 printd2$"**....................
...............**"d1$;
1210 printd1$"*.....................
................*"d2$;
1220 printd2$"**....................
...............**"d1$;
1230 printd1$"*......(c) 1986..by j
. lellesch......*"d2$;
1240 printd2$"**....................
...............**"d1$;
1250 printd1$"*.....................
................*"d2$;
1270 printd2$"*********************
*****************"d1$;
1280 printq3$;
1290 fori=0to5000:nexti
1300 rem **************** menue *
1310 x=400
1320 do
1330 sound1,x,1:x=x+5
1340 loopuntilx=900
1350 sound1,900,50
1360 scnclr:color 1,3:color 0,1:col
or 4,1
1370 print:poke 194,1
1380 print"...............m e n u e
................."
1390 poke 194,0:print
1400 print:print
1410 print".......1....- klavier"
1420 print
1430 print".......2....- specialeff
ekte"
1440 print
1450 print".......3....- kombinatio
n"
1460 print
1470 print".......4....- powerplay"
1480 print
1490 print".......5....- joystick s
ound"
1500 print:print
1510 print"  bitte taste druecken "
1520 x=900
1530 do
1540 sound1,x,1:x=x-5
1550 loopuntilx=400
1560 getkeya$
1570 ifa$="1"thensound1,169,5:goto1
640
1580 ifa$="2"thensound1,596,5:goto2
250
1590 ifa$="3"thensound1,810,5:goto3
940
1600 ifa$="4"thensound1,917,5:goto4
800
1610 ifa$="5"thensound1,999,5:goto6
870
1620 goto1560
1630 rem ************ klavier *
1640 scnclr :color 1,6:print:poke 1
94,1
1650 print
```

```
1660 print"..............klavier...
................"
1670 poke 194,0
1680 print:print
1690 input"...lautstaerke (1-8)..";
a
1700 ifa<1ora>8then1640
1710 vola
1720 print:print
1730 print"spielen sie mit den tast
en :"
1740 print
1750 print"  q bis '+' "
1760 print
1770 fori=1to1000:nexti
1780 goto2060
1790 getkeym$
1800 ifm$="q"thens=169
1810 ifm$="w"thens=262
1820 ifm$="e"thens=345
1830 ifm$="r"thens=383
1840 ifm$="1"thens=215
1850 ifm$="t"thens=453
1860 ifm$="2"thens=303
1870 ifm$="y"thens=516
1880 ifm$="4"thens=418
1890 ifm$="u"thens=571
1900 ifm$="5"thens=485
1910 ifm$="i"thens=596
1920 ifm$="6"thens=543
1930 ifm$="o"thens=643
1940 ifm$="8"thens=608
1950 ifm$="p"thens=685
1960 ifm$="9"thens=664
1970 ifm$="§"thens=704
1980 ifm$=":"thens=722
1990 ifm$="+"thens=739
2000 ifm$=";"thens=754
2010 ifm$="*"thens=784
2020 ifm$="s"then2710
2030 if m$="m"then1360
2040 sound1,s,20:sound2,169,20
2050 goto1790
2060 scnclr
2070 printspc(250):printspc(110):pr
int"
B..B..B..B..B..B..B..B..B..B..B..B.
.B"
2080 color 1,1:color 0,2:color 4,2
2090 print" B..B..B..B..B..B..B..B.
.B..B..B..B..B"
2100 print" B..B..B..B..B..B..B..B.
.B..B..B..B..B"
2110 print" B..B..B..B..B..B..B..B.
.B..B..B..B..B"
2120 print" Bq Bw Be Br Bt By Bu Bi
 Bo Bp B§ B+ B"
2130 print"
FCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFCCFC
CF"
2140 print chr$( 19)
2150 printspc( 80):print" #  #  B  #
 #  #  B  #  #  B  #  #  #
2160 print" #  #  B  #  #  #  B  #
 #  B  #  #  #
2170 print" #  #  B  #  #  #  B  #
 #  B  #  #  #
2180 print" #  #  B  #  #  #  B  #
 #  B  #  #  #
2190 print" #  #  B  #  #  #  B  #
 #  B  #  #  #
2200 print" 1  2  B  4  5  6  B  8
 9  B  :  ;  * "
2210 print" B  B  B  B  B  B  B  B
 B  B  B  B  B "
2220 print chr$( 19)
2230 print " s = ende.....m = menu
e"
2240 goto1790
2250 scnclr
2260 rem ******* spezialeffekte *
2270 color 1,11,6:color 0,1:color 4
,1
2280 print
2290 poke 194,1
2300 print"..............spezialeffe
kte............"
2310 poke 194,0
2320 print:print
2330 print" spielen sie mit den tas
ten 1 bis 8 "
2340 print" die spezialeffekte aus
!!!"
2350 print:print
2360 print"...bitte warten "
2370 fori=1to2500:nexti
2380 scnclr
2390 print
2400 print".....1...=..tief > hoch"
2410 print
2420 print".....2...=..hoch < tief"
2430 print
2440 print".....3...=..tief + rausc
h"
2450 print
2460 print".....4...=..hoch + rausc
h"
2470 print
2480 print".....5...=..tief + hoch"
2490 print
2500 print".....6...=..rausch > rau
sch"
2510 print
2520 print".....7...=..rausch < rau
sch"
2530 print
```

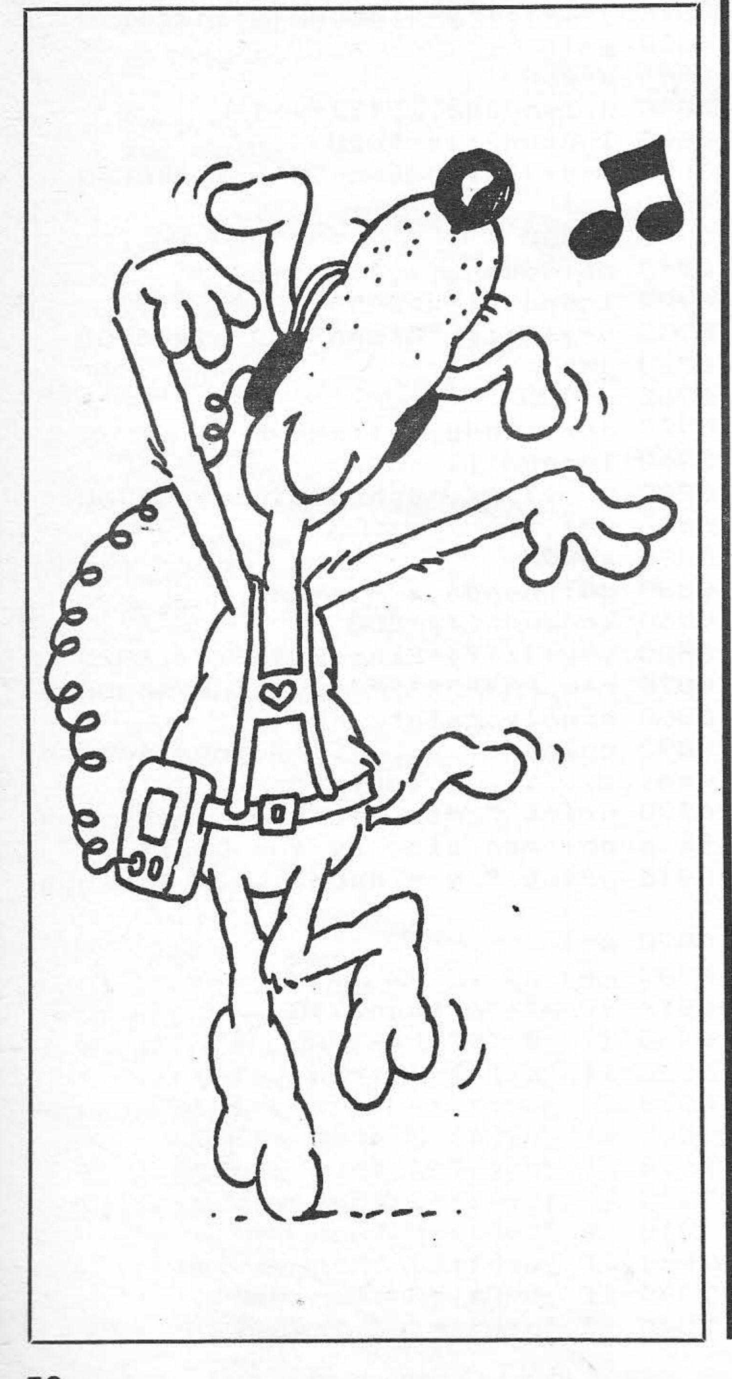
```
2540 print".....8...=..dreiklang (h
intereinander)"
2550 print
2560 print".....e...=..ende"
2570 print
2580 input"...bitte taste druecken
";a$
2590 ifa$="1"then2850
2600 ifa$="2"then3000
2610 ifa$="3"then3140
2620 ifa$="4"then3250
2630 ifa$="5"then3370
2640 ifa$="6"then3460
2650 ifa$="7"then3560
2660 ifa$="8"then3650
2670 ifa$="9"thenrun
2680 ifa$="0"thenrun
2690 ifa$="e"then2710
2700 goto2580
2710 scnclr:char 1,1,10,"..........
...e n d e"
2720 color 1,9:color 0,7,5:color4,7
,3
2730 char 1,1,15,"............good
bye "
2740 vol6
2750 sound1,810,30:sound1,834,30
2760 sound1,854,30:sound1,864,30
2770 sound1,881,30:sound1,897,30
2780 sound1,911,30:sound1,917,275
2790 x=500
2800 do:sound2,x,5:x=x-10
2810 loopuntilx<20
2820 color1,1:color0,2:color 4,13,5
2830 scnclr
2840 end
2850 rem ************ tief > hoch *
2860 scnclr
2870 print
2880 print"...............tief > ho
ch  "
2890 print:print
2900 input"...tiefste zahl ";a
2910 input"  hoechste zahl ";b
2920 input"  schrittweite (1-50) ";
c
2930 print:print"..okay "
2940 rem ************* schleife *
2950 sound1,a,5:a=a+c
2960 ifa>bthen2970:else2940
2970 print:print
2980 input" menue (m) oder speziale
. (s)";s$
2990 ifs$="m"then1360:else2380
3000 scnclr
3010 rem ********** hoch < tief *
3020 print
3030 print"...............hoch < ti
ef  "
3040 print:print
3050 input"  hoechste zahl ";d
3060 input"  tiefste zahl ";e
3070 input"  schrittweite (1-50) ";
f
3080 print:print"  okay  "
3090 rem ************* schleife *
3100 sound1,d,5:d=d-f
3110 ifd<fthen3120:else3090
3120 print:print
3130 goto2980
3140 scnclr
3150 rem ******** tief + rausch *
3160 print"............tief + raus
ch "
3170 print:print
3180 input" rausch zahl (1-1000)
";g
3190 print:print"  okay  "
3200 sound3,g,600
3210 x=1
3220 do:sound1,x,5:x=x+5
3230 loopuntilx>600
3240 print:print:goto2980
3250 scnclr
3260 rem ******** hoch + rausch *
3270 print
3280 print"...............hoch + ra
usch"
3290 print:print
3300 input" rausch zahl (1-1000) ";
h
3310 print:print"  okay  "
3320 sound3,h,600
3330 x=600
3340 do:sound1,x,5:x=x-5
3350 loopuntilx<10
3360 print:print:goto2980
3370 scnclr
3380 rem ********** tief + hoch *
3390 print
3400 print"..............tief + hoc
h.."
3410 print:print
3420 input"hoechster ton ";i
3430 input"tiefster ton ";j
3440 sound1,i,100:sound2,j,100
3450 print:print:goto2980
3460 scnclr
3470 rem ****** rausch > rausch *
3480 print"...............rausch >
rausch "
3490 print:print
3500 input" hoechste zahl ";k
3510 input" kleinste zahl ";l
3520 do
3530 sound3,k,5:k=k-5
```

```
3540 loopuntilk<1
3550 print:print:goto2980
3560 scnclr
3570 rem ****** rausch < rausch *
3580 print"..............rausch <
rausch  "
3590 print:print
3600 input" kleinste zahl ";m
3610 input" hoechste zahl ";n
3620 do:sound3,m,5:m=m+5
3630 loopuntilm>n
3640 print:print:goto2980
3650 scnclr
3660 rem ************ dreiklang *
3670 print
3680 print"..............dreiklang
.."
3690 print:print
3700 input" c,d,e,f,g,a,h, welcher
";o$
3710 ifo$="c"thenz=810
3720 ifo$="d"thenz=834
3730 ifo$="e"thenz=854
3740 ifo$="f"thenz=864
3750 ifo$="g"thenz=881
3760 ifo$="a"thenz=897
3770 ifo$="h"thenz=911
3780 ifz=810thenx=834
3790 ifz=810theny=854
3800 ifz=834thenx=854
3810 ifz=834theny=864
3820 ifz=854thenx=864
3830 ifz=854theny=881
3840 ifz=864thenx=881
3850 ifz=864theny=897
3860 ifz=881thenx=897
3870 ifz=881theny=911
3880 ifz=897thenx=911
3890 ifz=897theny=929
3900 ifz=911thenx=929
3910 ifz=911theny=939
3920 sound1,z,30:sound1,x,30:sound1
,y,45
3930 print:print:goto2980
3940 scnclr
3950 rem ********** kombination *
3960 color 1,16:color 0,1:color 4,1
3970 poke 194,1
3980 print".............kombinatio
n............."
3990 poke 194,0
4000 print:print
4010 print" druecken sie mehrere ta
sten hinter -"
4020 print" einander !!!"
4030 print:print"  bitte warten "
4040 fori=1to2000:nexti
4050 scnclr
4060 print
4070 print"........q...= tief < ho
ch"
4080 print
4090 print"........w...= hoch > ti
ef"
4100 print
4110 print"........e...= rau. + ti
ef"
4120 print
4130 print"........r...= rau. + ho
ch"
4140 print
4150 print"........t...= rau. > ra
u."
4160 print
4170 print"........y...= rau. < ra
u."
4180 print
4190 print"........u...= hell...ra
u."
4200 print
4210 print"........i...= dunk...ra
u."
4220 print
4230 print"........o...= mitl...ra
u."
4240 print
4250 print"........m...= menue"
4260 print
4270 print"........s...= ende"
4280 print
4290 print" bitte taste druecken  "
4300 getkeya$
4310 ifa$="q"then4440
4320 ifa$="w"then4500
4330 ifa$="e"then4560
4340 ifa$="r"then4590
4350 ifa$="t"then4620
4360 ifa$="y"then4670
4370 ifa$="u"then4720
4380 ifa$="i"then4750
4390 ifa$="o"then4780
4400 ifa$="m"then1300
4410 ifa$="s"then2710
4420 goto4300
4430 :
4440 x=1
4450 do
4460 sound1,x,5:x=x+15
4470 loopuntilx>1000
4480 goto4300
4490 :
4500 x=1000
4510 do
4520 sound1,x,5:x=x-15
4530 loopuntilx<100
4540 goto4300
```

```
4550 :
4560 sound1,100,75:sound3,789,75
4570 goto4300
4580 :
4590 sound1,890,75:sound3,100,75
4600 goto4300
4610 :
4620 x=1000
4630 do:sound3,x,5:x=x-15
4640 loopuntilx<30
4650 goto4300
4660 :
4670 x=1
4680 do:sound3,x,5:x=x+15
4690 loopuntilx>1000
4700 goto4300
4710 :
4720 sound1,888,75:sound3,999,75
4730 goto4300
4740 :
4750 sound1,1,75:sound3,1,75
4760 goto4300
4770 :
4780 sound1,567,75:sound3,567,75
4790 goto4300
4800 :
4810 rem ************ powerplay *
4820 :
4830 color 1,13,6:color 0,1:color 4
,1
4840 scnclr
4850 poke 194,1
4860 print"............powerplay...
................"
4870 poke 194,0
4880 print:print
4890 print"..die tasten..a bis z..s
ind alle mit"
4900 print" verschiedenen geraeusch
kombinationen.....belegt !"
4910 print" sie koennen zwischen 2
modis waehlen"
4920 print" rauschen - (a)
4930 print" ton      - (b)
4940 getkeya$
4950 ifa$="a"thena=3
4960 ifa$="b"thena=1
4970 ifa$<>"a"anda$<>"b"then4940
4980 print:print" ende   = 3 "
4990 print" menue = 4"
5000 print" bitte taste druecken  "
5010 getkeya$
5020 ifa$="a"then5310
5030 ifa$="b"then5370
5040 ifa$="c"then5430
5050 ifa$="d"then5490
5060 ifa$="e"then5550
5070 ifa$="f"then5610
5080 ifa$="g"then5670
5090 ifa$="h"then5730
5100 ifa$="i"then5790
5110 ifa$="j"then5850
5120 ifa$="k"then5910
5130 ifa$="l"then5970
5140 ifa$="m"then6030
5150 ifa$="n"then6090
5160 ifa$="o"then6150
5170 ifa$="p"then6210
5180 ifa$="q"then6270
5190 ifa$="r"then6330
5200 ifa$="s"then6390
5210 ifa$="t"then6450
5220 ifa$="u"then6510
5230 ifa$="v"then6570
5240 ifa$="w"then6630
5250 ifa$="x"then6690
5260 ifa$="y"then6750
5270 ifa$="z"then6810
5280 ifa$="3"then2710
5290 ifa$="4"then1300
5300 goto5010
5310 y=1
5320 x=1020
5330 do:sounda,x,1:x=x-30
5340 loop untilx<800
5350 y=y+1:ify=5then5010:else5320
5370 y=1
5380 x=1020
5390 do:sounda,x,3:x=x-50
5400 loop untilx<800
5410 y=y+1:ify=5then5010:else5380
5430 y=1
5440 x=1020
5450 do:sounda,x,3:x=x-20
5460 loopuntilx<800
5470 y=y+1:ify=3then5010:else5440
5490 y=1
5500 x=1020
5510 do:sounda,x,1:x=x-90
5520 loopuntilx<800
5530 y=y+1:ify=8then5010:else5500
5550 y=1
5560 x=1020
5570 do:sounda,x,10:x=x-80
5580 loopuntilx<800
5590 y=y+1:ify=4then5010:else5560
5610 y=1
5620 x=800
5630 do:sounda,x,1:x=x+20
5640 loopuntilx=1020
5650 y=y+1:ify=5then5010:else5620
5670 y=1
5680 x=800
5690 do:sounda,x,1:x=x+5
5700 loopuntilx=1020
5710 y=y+1:ify=3then5010:else5680
```

```
5730 y=1
5740 x=800
5750 do:sounda,x,1:x=x+55
5760 loopuntilx=1020
5770 y=y+1:ify=7then5010:else5740
5790 y=1
5800 x=800
5810 do:sounda,x,6:x=x+20
5820 loopuntilx=1020
5830 y=y+1:ify=4then5010:else5800
5850 y=1
5860 x=800
5870 do:sounda,x,4:x=x+70
5880 loopuntilx=1010
5890 y=y+1:ify=8then5010:else5860
5910 y=1
5920 x=600
5930 do:sounda,x,1:x=x+50
5940 loopuntilx=1000
5950 y=y+1:ify=8then5010:else5920
5970 y=1
5980 x=600
5990 do:sounda,x,4:x=x+25
6000 loopuntilx=1000
6010 y=y+1:ify=5then5010:else5980
6030 y=1
6040 x=600
6050 do:sounda,x,5:x=x+75
6060 loopuntilx>940
6070 y=y+1:ify=5then5010:else6040
6090 y=1
6100 x=1000
6110 do:sounda,x,5:x=x-75
6120 loopuntilx<700
6130 y=y+1:ify=5then5010:else6100
6150 y=1
6160 x=1000
6170 do:sounda,x,1:x=x-30
6180 loopuntilx<600
6190 y=y+1:ify=5then5010:else6160
6210 y=1
6220 x=800
6230 do:sounda,x,4:x=x-30
6240 loopuntilx<450
6250 y=y+1:ify=5then5010:else6220
6270 y=1
6280 x=450
6290 do:sounda,x,4:x=x+30
6300 loopuntilx>800
6310 y=y+1:ify=5then5010:else6280
6330 y=1
6340 x=1020
6350 do:sounda,x,4:x=x-30
6360 loopuntilx<600
6370 y=y+1:ify=5then5010:else6340
6390 y=1
6400 x=1020
6410 do:sounda,x,1:x=x-1
6420 loopuntilx<980
6430 y=y+1:ify=3then5010:else6400
6450 y=1
6460 x=1020
6470 do:sounda,x,4:x=x-1
6480 loopuntilx<995
6490 y=y+1:ify=3then5010:else6460
6510 y=1
6520 x=1020
6530 do:sounda,x,2:x=x-3
6540 loopuntilx<980
6550 y=y+1:ify=6then5010:else6520
6570 y=1
6580 x=980
6590 do:sounda,x,2:x=x+3
6600 loopuntilx>1016
6610 y=y+1:ify=5then5010:else6580
6630 y=1
6640 x=950
6650 do:sounda,x,1:x=x+1
6660 loopuntilx=1020
6670 y=y+1:ify=3then5010:else6640
6690 y=1
6700 x=800
6710 do:sounda,x,1:x=x+5
6720 loopuntilx>900
6730 y=y+1:ify=5then5010:else6700
6750 y=1
6760 x=900
6770 do:sounda,x,1:x=x-5
6780 loopuntilx<800
6790 y=y+1:ify=5then5010:else6760
6810 y=1
6820 x=900
6830 do:sounda,x,4:x=x-5
6840 loopuntilx<800
6850 y=y+1:ify=5then5010:else6820
6870 rem ********* joystick sound *
6880 scnclr:print
6890 print "...........joystick so
und............:print
6900 print " der joystick macht mus
ik.probieren sie  es aus (port 1)"
6910 print " e = ende.....m = menue
"
6920 x=1
6930 get a$
6940 if a$="e"then2710
6950 if a$="m"then1300
6960 if joy(1)=1 then y=169
6970 if joy(1)=2 then y=262
6980 if joy(1)=3 then y=345
6990 if joy(1)=4 then y=383
7000 if joy(1)=5 then y=453
7010 if joy(1)=6 then y=516
7020 if joy(1)=7 then y=571
7030 if joy(1)=8 then y=596
7040 if joy(1)=128 then 7070
```

```
7050 sound x,y,20
7060 goto6930
7070 if x=1 then x=3:goto 6930
7080 if x=3 then x=1
7090 goto6930
9000 rem supersound =============p4
9010 rem 60671 bytes memory     ====
9020 rem 12141 bytes program    ====
9030 rem 00000 bytes variables ====
9040 rem 00000 bytes arrays     ====
9050 rem 00000 bytes strings    ====
9060 rem 48530 bytes free       ====
9070 rem bei c-16 und c-116 ohne
9080 rem kopf und fusszeilen
9090 rem eingeben >speicher<
```



```
10 rem kfz wirtschaftlichkeit  =c16
20 rem (p)       commodore welt   =
30 rem =============================
40 rem (c)        by              =
50 rem hartm. schulteis           =
60 rem                            =
70 rem                            =
80 rem c16/c116/plus4 + 1531      =
90 rem =============================

100 deffnau(x)=int((x*100)+0.5)/100
110 cd$=chr$(17)
120 rn$=chr$(18):re$=chr$(28)
130 gr$=chr$(30):bl$=chr$(31)
140 bk$=chr$(144):rf$=chr$(146)
150 cl$=chr$(147):pu$=chr$(156)
160 forq=1to3:c3$=c3$+cd$:nextq
170 cy$=chr$(159):gosub2140
180 printcl$bl$;:char1,4,3,"dieses
programm berechnet die "
190 char1,4,5,"wirtschaftlichkeit d
es wagens :"
200 printre$rn$;:char1,15,7,"ac - h
s 49"
210 printbl$;:char1,15,11,"wagentyp
 :"
220 printgr$rn$;:char1,11,13,"ford
taunus tunier"
230 printre$;:char1,17,20,"taste"
240 geta$:ifa$=""then240
250 ifer%=1then2230
260 printcl$bk$"datum ";:inputda$
270 printpu$:char1,9,5,"haben sie g
etankt ?":
280 print:print:print:printtab(8)"
"gr$rn$"n"rf$"ein/"rn$"v"rf$"oll/"r
n$"t"rf$"eilweise"
290 geta$:ifa$<>"n"anda$<>"v"anda$<
>"t"then290
300 ifa$="n"thenkd%=1:goto410
310 ifa$="t"thenkd%=1:goto340
320 printre$:char1,8,11,"bei kilome
terstand :"
330 inputks:printcd$bk$:goto350
340 printcd$bk$
350 printtab(8);:input"wieviel lite
r ";li:il=li
360 printcd$cy$
370 printtab(8):input"wieviel dm ";
dx
380 printbl$:char1,8,22,"stimmt all
es ?  (j/n)"
390 geta$:ifa$<>"j"anda$<>"n"then39
0
400 ifa$="n"thenkd%=0:goto260
410 printcl$re$rn$;:char1,5,2,"wirt
schaftlichkeitsberechnung"
420 printgr$rn$;:char1,7,5,"1":prin
```

```
tbk$rf$;:char1,11,5,"durchschnittsv
erbrauch"
430 printgr$rn$;:char1,7,7,"2":prin
tbk$rf$;:char1,11,7,"tankkalkulatio
n"
440 printgr$rn$;:char1,7,9,"3":prin
tbk$rf$;:char1,11,9,"wagenkosten"
450 printgr$rn$;:char1,7,11,"4":pri
ntbk$rf$;:char1,11,11,"daten laden"
460 printgr$rn$;:char1,7,13,"5":pri
ntbk$rf$;:char1,11,13,"daten speich
ern"
470 printgr$rn$;:char1,7,15,"6":pri
ntbk$rf$;:char1,11,15,"ende"
480 printpy$rn$;:char1,7,17,"bitte
waehlen sie"
490 remgetpf$:ifpf$=""then495
500 rem pf=val(pf$)
510 pf=pf+1
520 ifpf=1then570
530 ifpf=2then600
540 ifpf=3then630
550 ifpf=4then660
560 gosub670:goto870
570 poke3512,45:poke3513,62:poke327
2,81
580 poke3352,81:poke3432,81:poke359
2,81
590 goto730
600 ifkd%=1then510
610 poke3272,45:poke3273,62:poke335
2,81:poke3592,81
620 gosub670:goto760
630 gosub670:ifpeek(3432)=86then510
640 poke3432,45:poke3433,62:poke359
2,81
650 goto810
660 poke3592,45:poke3593,62:gosub67
0:goto870
670 ifak%=1thenpoke3272,86
680 ifbk%=1thenpoke3352,86
690 ifck%=1thenpoke3432,86
700 ifdk%=1thenpoke3512,86:ifdv=0th
enpoke3272,81:poke3352,81
710 ifek%=1thenpoke3592,86
720 return
730 geta$:ifa$<>"4"anda$<>"6"then73
0
740 ifa$="4"then1750
750 goto1130
760 geta$:ifa$<>"1"anda$<>"3"anda$<
>"4"anda$<>"6"then760
770 ifa$="1"then1410
780 ifa$="3"then1210
790 ifa$="4"then1750
800 goto1130
810 geta$:ifa$<>"1"anda$<>"2"anda$<
>"3"anda$<>"4"anda$<>"6"then810
820 ifa$="1"then1410
830 ifa$="2"then910
840 ifa$="3"then1210
850 ifa$="4"then1750
860 goto1130
870 geta$:ifa$=""then870
880 ifa$<>"1"anda$<>"2"anda$<>"3"an
da$<>"4"anda$<>"5"anda$<>"6"then870
890 a%=val(a$)
900 ona%goto1410,910,1210,1750,1540
,1130
910 printcl$re$:char1,12,2,"tankkal
kulation"
920 printbl$:char1,4,5,"was haetten
 sie gerne berechnet ?"
930 print:print:printpu$tab(7)rn$"k
"rf$"m / "rn$"l"rf$"iter"
940 geta$:ifa$<>"k"anda$<>"l"then94
0
950 ifa$="l"then1020
960 printgr$:char1,5,10,"fuer wievi
el liter"
970 inputi
980 ak=(i/gd)*100
990 ak=fnau(ak)
1000 print:print:printbk$"damit kom
men sie ca. "ak"km weit."
1010 goto1100
1020 printgr$:char1,5,14,"fuer wiev
iel km"
1030 inputkn
1040 al=(gd/100)*kn
1050 al=fnau(al)
1060 ap=lp*al
1070 ap=fnau(ap)
1080 printcd$bk$"sie brauchen dafue
r ca."al"liter."
1090 printcd$"( ="ap"dm )"
1100 printyu$:char1,16,22,"taste"
1110 geta$:ifa$=""then1110
1120 pf=pf-1:bk%=1:goto410
1130 ifek%=1then1200
1140 printcl$bk$;:char1,0,3,"das wa
ere aber nicht klug,da sie die da-"
1150 printcd$"ten,die fuer die stat
istik wichtig sind,"cd$tab(10)"nich
t speichern !!!"
1160 printpu$;:char1,4,16,"wollen s
ie noch einmal zurueck ?"
1170 printcy$;:char1,15,22,"( j / n
 )"
1180 geta$:ifa$<>"j"anda$<>"n"then1
180
1190 ifa$="j"thenpf=pf-1:goto410
1200 printcl$bk$:color0,2:color4,7,
6:end
1210 printcl$re$tab(14)"wagenkosten
"
```

```
1220 printcd$cd$bk$tab(9)"haben sie
 nebenkosten?"
1230 printcd$cd$yu$tab(15)"( j / n
)"
1240 gp=aa:gk=bb:gp=gp+dx
1250 geta$:ifa$<>"j"anda$<>"n"then1
250
1260 ifa$="n"then1300
1270 printcd$gr$"nebenkosten :"cd$
1280 input"dm ";nk
1290 gp=gp+nk:goto1310
1300 printcd$cd$
1310 ifgk=0then1340
1320 pk=gp/gk
1330 pk=fnau(pk)
1340 printcd$bl$"ihr kfz. kostet ih
nen pro km :"pk"dm"
1350 printcd$cy$"ihnen hat der wage
n schon"gp"dm"
1360 printcd$"ingesamt gekostet !!!
"
1370 printpu$:char1,16,22,"taste"
1380 geta$:ifa$=""then1380
1390 ifck%=1thenpf=pf-1
1400 ck%=1:goto410
1410 printcl$re$;:char1,9,1,"durchs
chnittsverbrauch"
1420 printbk$:char1,3,3,"der durchs
chnittsverbrauch betraegt :"
1430 print:print:printdv:char1,10,5
,"liter auf 100 km"
1440 ifkd%=1orak%=1then1460
1450 gd=gd+dv:ei=ei+1
1460 gv=gd/ei
1470 gv=fnau(gv)
1480 print:print:print:printgr$"der
 gesamtdurchschnitt liegt bei":prin
t
1490 printcd$gv"liter auf 100 km"
1500 printcy$:char1,16,22,"taste"
1510 geta$:ifa$=""then1510
1520 ifak%=1orkd%=1thenpf=pf-1
1530 ak%=1:goto410
1540 printcl$cd$re$tab(12)"daten sp
eichern"
1550 printbk$:char1,9,5,"datenband
einlegen!!!"
1560 printgr$:char1,16,10,"taste"
1570 geta$:ifa$=""then1570
1580 printpu$:char1,9,10,"daten wer
den gesichert"
1590 print:print:print
1600 open1,1,1
1610 print#1,gp;chr$(13);
1620 print#1,gk;chr$(13);
1630 print#1,gd;chr$(13);
1640 print#1,ei;chr$(13);
1650 print#1,da$;chr$(13);
1660 print#1,ks;chr$(13);
1670 print#1,li;chr$(13);
1680 print#1,dv;chr$(13);
1690 print#1,il;chr$(13);
1700 print#1,dx;chr$(13);
1710 print#1,kk;chr$(13);
1720 close1
1730 ifff=0then1200
1740 ek%=1:goto410
1750 printcl$re$;:char1,14,3,"daten
 laden"
1760 printbk$;:char1,9,5,"datenband
 einlegen!!!"
1770 printgr$:char1,16,10,"taste"
1780 geta$:ifa$=""then1780
1790 printpu$;:char1,10,10,"daten w
erden geladen":print:print
1800 open1,1,0
1810 input#1,aa,bb,cc,dd,ee$,kk,gg,
hh,ii,jj,kk
1820 close1
1830 printcl$bk$"diese angaben sind
 vom..."ee$
1840 printcd$cd$re$"seit verwendung
 dieses programms,...":printcd$"sin
d sie"
1850 printcd$bb"km gefahren."
1860 printc3$bl$"ihr wagen hat sie
bis zu diesem zeit- ":printcd$"punk
t"
1870 printcd$aa"dm gekostet."
1880 printgr$:char1,16,22,"taste"
1890 geta$:ifa$=""then1890
1900 printcl$cd$bk$"der letzte durc
hschnittsverbrauch lag":print
1910 printcd$"bei"hh"liter auf"
1920 printcd$"100 km."
1930 lp=jj/ii
1940 lp=fnau(lp)
1950 printcd$cd$gr$"der kraftstoffl
iterpreis betrug"lp"dm."
1960 printcd$cd$re$"nun geben sie w
eitere daten ein!!!"
1970 printcy$:char1,16,22,"taste"
1980 ifkd%=1then2040
1990 li=li+gg
2000 di=ks-ff:bb=bb+di
2010 dv=(li/di)*100
2020 dv=fnau(dv)
2030 li=0:kk=0:goto2070
2040 ifkk=0thenkk=1:goto2060
2050 li=li+gg
2060 dv=hh:ks=ff
2070 gd=cc:ei=dd
2080 ifds=0thenil=ii:dx=jj:goto2110
2090 lp=dx/il
2100 lp=fnau(lp)
2110 geta$:ifa$=""then2110
```

```
2120 ifdk%=1thenpf=1
2130 dk%=1:goto410
2140 color0,2,5:color1,2,5
2150 printcl$bl$rn$:char1,5,3,"wirt
schaftlichkeitsberechnung"
2160 printgr$:char1,16,5,"(c) 1985"
2170 char1,11,5,"hartmut schulteis"
2180 printpu$:char1,0,8,"kurze erla
euterungen zu diesem programm?"
2190 printcy$:char1,15,20,"( j / n
)"
2200 geta$:ifa$<>"j"anda$<>"n"then2
200
2210 ifa$="n"thenreturn
2220 er%=1:return
2230 printcl$re$" benutzen sie das
programm zum 1. mal ? ";
2240 printcd$cd$cy$tab( 16)"( j / n
)"
2250 geta$:ifa$<>"j"anda$<>"n"then2
250
2260 ifa$="n"then2370
2270 printcd$cd$bk$"datum ";:inputd
a$
2280 printcd$cd$gr$"bei kilometerst
and :"cd$
2290 inputks:printcd$bk$
2300 input"wieviel liter ";il
2310 printcd$cy$
2320 input"wieviel dm ";dx
2330 printcd$cd$bl$"........stimmt
alles ?..( j/n)"
2340 geta$:ifa$<>"j"anda$<>"n"then2
340
2350 ifa$="n"thenprintcl$cd$cd$:got
o2270
2360 goto1540
2370 rem erlaeuterungen
2380 printcl$re$"..."rn$"erlaeuteru
ngen"rf$
2390 printc3$gr$"es wird berechnet
gesamt- und tankfuel-"
2400 printcd$"lungdurchschnittsverb
rauch, tankkalkula-"
2410 printcd$"tion,wagenkosten insg
esamt und pro km":printcd$"und der
literpreis"
2420 printc3$bl$tab( 18)"taste"
2430 geta$:ifa$=""then2430
2440 printcl$pu$"symbolerklaerung :
"
2450 printcd$bl$"->"bk$" tipvorschl
ag vom computer"
2460 printcd$pu$"Q"bk$" nicht anwa
ehlbar"
2470 printcd$cy$"V"bk$" schon ange
waehlt"
2480 printcd$gr$"sie koennen einen
programmteil nochmal":printcd$
2490 printcd$"anrufen, wenn sie gla
uben,eine korrektur"
2500 printcd$"vornehmen zu muessen.
"
2510 printcd$cd$bl$tab( 18)"taste"
2520 geta$:ifa$=""then2520
2530 goto260
2540 rem kfz wirtschaftlichkeit c16
2550 rem 19967 bytes memory       ==
2560 rem 07374 bytes programm     ==
2570 rem 00000 bytes variables    ==
2580 rem 00000 bytes arrays       ==
2590 rem 00000 bytes strings      ==
2600 rem 12593 bytes free (0)     ==
2610 rem ===========================
```

# FOTO

Ein Programm für den „Profi-" und Hobbyfotografen, der mit den Begriffen Schärfentiefe, Entfernungsblendeneinstellung, Abbildungsmaßstab, Vorsatzlinsen, Farbtemperatur, Blitzberechnung und Tabellen etwas anfangen kann.
Aus diesen Menüpunkten kann zu Beginn des Programmes ausgewählt werden. Sodann fragt das Programm nach den entsprechenden Daten, z.B. Brennweite, und rechnet dann die gewünschten Ergebnisse selbständig aus. Weiterer Erklärung bedarf es eigentlich nicht, dem fotoversierten Anwender erklärt sich dieses Programm von selbst.

# ENGLISCHE VERBEN

Das Programm gibt Ihnen ein englisches Verb in der deutschen Übersetzung. Sie müssen nun der Reihe nach dessen Stammformen (natürlich in Englisch) eingeben. Nach jeder Eingabe muß >RETURN< gedrückt werden. Sind alle drei Stammformen eingegeben, gibt der C-16 die Musterlösung auf dem Bildschirm aus. Für ein neues Verb muß eine beliebige Taste gedrückt werden, dann wird ein neues Verb abgefragt. Nach Durchlauf aller Verben erstellt der C-16 Ihre Fehlerquote nach den zu erreichenden Punkten (685) und errechnet die Schulnote.

```
10 rem foto=================plus 4=
20 rem (p) commodore welt team   ==
30 rem ============================
40 rem (c) by                     =
50 rem bernd welte                =
60 rem                            =
70 rem 40z/ascii                  =
80 rem plus 4/c166+ 1571/1541/1530=
90 rem ============================

100 printchr$(014)
110 c4$=chr$(017):cl$=chr$(147)
120 re$=chr$(028):bl$=chr$(031)
130 rn$=chr$(018):rf$=chr$(146)
140 zz$=chr$(042)
150 forq=1to10:qa$=qa$+c4$:nextq
160 forq=1to3:q3$=q3$+c4$:nextq
170 forq=1to40:qb$=qb$+zz$:nextq
180 ze=2035:sp=2036:cu=65520
190 wd$="Weitere Daten ? (j/n)"
200 wt$="Bitte warten"
210 bw$="Bitte waehlen Sie"
220 printcl$qb$;
230 print"*.........................
..............*";
240 print"*..........A&B Welte Soft
ware .........*";
250 print"*.........................
..............*";
260 printqb$;
270 print
280 print
290 print" (1) Schaerfentiefe..."
300 print
310 print" (2) Entfernungs -Blenden
 Einstellung"
320 print
330 print" (3) Abbildungsmassstab"
340 print
350 print" (4) Vorsatzlinsen...."
360 print
370 print" (5) Farbtemperatur..."
380 print
390 print" (6) Blitzberechnung.."
400 print
410 print" (7) Tabellen........."
420 print
430 print" (8) Ende............"
440 print
450 print"....Bitte waehlen sie das
 Modul aus"
460 geta$:ifa$=""then460
470 ifa$="1"then560
480 ifa$="2"then1080
490 ifa$="3"then1310
500 ifa$="4"then1760
510 ifa$="5"then2650
520 ifa$="6"then1980
530 ifa$="7"then3010
540 ifa$="8"thenend
550 goto460
560 printcl$
570 print".........Schaerfentiefent
abelle.........";
580 print"..Zerstreuungskreisdurchm
esser in mm":print:inputc$
590 c=val(c$)
600 c=c/1000
610 print:print"Brennweite in mm":p
rint:inputf$
620 f=val(f$)
630 f=f/1000
640 print:print"Gegenstandsweite in
 m":print:inputg$:printcl$
650 g=val(g$)
660 ifg>fthen710
670 printc4$re$"....Bedingung"bl$
680 printc4$rn$"...Gegenstandsweite
..>.als Brennweite...";
690 printrn$".............nicht erf
uellt!............";rf$
700 print" Neuer Wert":inputg$
710 g=val(g$)
720 w=sqr(2)
730 b=1
740 fori=1to10
750 h=f*f/(b*c)
760 k=h*g/(h+g-f)
770 ifh+f>gthen890
780 b=int((b+.05)*10)/10
790 h=int((h+.05)*1000)/1000
800 k=int((k+.05)*1000)/1000
810 l=int((l+.05)*1000)/1000
820 pokeze,3:pokesp,10:syscu
830 print"Blende";:printtab(28);:pr
intusing"##.##";b
840 pokeze,6:pokesp,4:syscu
850 print"Hyperfokale Entfernung ";
:printusing"###.####";h;:print" m"
860 printc4$".......minimale Entfer
nung ";:printusing"###.####";k;:pri
nt" m"
870 printc4$".......maximale Entfer
nung ";:printusing"###.####";l;:pri
nt" m"
880 goto1000
890 l=h*g/(h-g+f)
900 b=int((b+.05)*10)/10
910 h=int((h+.05)*1000)/1000
920 k=int((k+.05)*1000)/1000
930 l=int((l+.05)*1000)/1000
940 pokeze,3:pokesp,10:syscu
950 print"Blende";:printtab(28);:pr
intusing"##.##";b
960 pokeze,6:pokesp,4:syscu
970 print"Hyperfokale Entfernung ";
```

```
:printusing"###.####";h;:print" m"
980 printc4$".......minimale Entfer
nung ";:printusing"###.####";k;:pri
nt" m"
990 printc4$".......maximale Entfer
nung ";:printusing"###.####";l;:pri
nt" m"
1000 b=b*w
1010 char1,12,20,wt$
1020 fort=1to2000:next
1030 char1,12,20,wd$
1040 geta$:ifa$=""then1040
1050 ifa$="j"thenprintcl$:goto600
1060 ifa$="n"then220
1070 goto1040
1080 printcl$q3$" Optimale Entfernu
ngs Blendeneinstellung"
1090 printq3$"..Zerstreuungskreis D
urchmesser in mm"
1100 print:inputc$
1110 c=val(c$):c=c/1000
1120 printc4$" Brennweite in mm":in
putf$:print
1130 f=val(f$):f=f/1000
1140 print:print" Minimale Entfernu
ng in m":inputk$:k=val(k$)
1150 print:print" Maximale Entfernu
ng in m":inputl$:l=val(l$)
1160 g=2*k*l/(k+l)
1170 h=(2*k*l-f*(k+l))/(l-k)
1180 b=f*f/(h*c)
1190 g=int((g+.005)*100)/100
1200 h=int((h+.005)*100)/100
1210 b=int((b+.05)*10)/10
1220 printcl$c4$" Optimale Enfernun
g.....=";g"m":print
1230 print:print" Hyperfokale Entfe
rnung =";h"m":print
1240 print:print" Mindest Blendenwa
hl....=";b:print
1250 char1,10,20,wd$
1260 geta$:ifa$=""then1260
1270 ifa$="j"then1080
1280 ifa$="n"then220
1290 goto1260
1300 rem abbildungsmassstab
1310 printcl$q3$".........Abbildung
smassstab"
1320 printc4$"...Alle Masse bitte i
n mm eingeben."
1330 printc4$"......Was soll berech
net werden?"
1340 printc4$
1350 printc4$"..........(1) Abbildu
ngsmassstab"
1360 printc4$"..........(2) Auszugs
verlaengerung"
1370 printc4$"..........(3) Brennwe
ite"
1380 printc4$"..........(4) Hauptme
nue"
1390 char1,10,20,bw$
1400 geta$:ifa$=""then1400
1410 ifa$="1"then1460
1420 ifa$="2"then1560
1430 ifa$="3"then1660
1440 ifa$="4"then220
1450 goto1400
1460 printcl$q3$"  Abbildungsmassst
ab":print:print
1470 printc4$" Brennweite":print:in
putbr$:br=val(br$)
1480 printc4$" Auszugsverlaengerung
":print:inputas$:as=val(as$)
1490 ab=as/br
1500 printc4$" Abbildungsmassstab":
print:print:printtab(7);ab":1"
1510 char1,10,20,wd$
1520 geta$:ifa$=""then1510
1530 ifa$="j"then1460
1540 ifa$="n"then1300
1550 goto1510
1560 printcl$q3$" Auszugsverlaenger
ung":print:print
1570 printc4$" Brennweite":print:in
putbr$:br=val(br$)
1580 printc4$" Abbildungsmassstab":
print:inputab$:ab=val(ab$)
1590 as=br*ab
1600 printc4$" Auszugsverlaengerung
":print:printtab(7);as"mm"
1610 char1,10,20,wd$
1620 geta$:ifa$=""then1620
1630 ifa$="j"then1560
1640 ifa$="n"then1300
1650 goto1620
1660 printcl$q3$"....Brennweite....
..":print:print
1670 printc4$" Auszugsverlaengerung
":print:inputas$:as=val(as$)
1680 printc4$" Abbildungsmassstab":
print:inputab$:ab=val(ab$)
1690 br=as/ab
1700 printc4$" Brennweite":print:pr
inttab(7);br"mm"
1710 char1,10,20,wd$
1720 geta$:ifa$=""then1720
1730 ifa$="j"then1660
1740 ifa$="n"then1300
1750 goto1720
1760 printcl$
1770 print".........Linsenkombinat
ionen":print
1780 printc4$"Aus Objektivbrennweit
e und Vorsatzlinse"
1790 printc4$"....wird das Gesamtsy
```

```
stem errechnet."
1800 char1,10,20,wt$
1810 fort=1to2000:next
1820 printchr$(147)
1830 print:print" Brennweite in mm"
:print:inputf
1840 print:print" Brechkraft in Dio
ptr.":print:inputd
1850 f1=(1/f)*1000:f2=f1
1860 gb=f1+d:gf=1/gb:d2=1/d
1870 printcl$
1880 print:print" Brennweite.....";
f"mm =";f2"Dioptrien"
1890 d2=int((d2+.0005)*1000)/1000
1900 print:print" Brechkraft.....";
d"Dioptrien =";d2"mm"
1910 gf=int((gf+.0005)*1000)/1000
1920 print:print" Gesamtsystem...";
gb"Dioptr.=";gf"mm"
1930 char1,10,20,wd$
1940 geta$:ifa$=""then1940
1950 ifa$="j"then1820
1960 ifa$="n"then220
1970 goto1940
1980 printcl$:print"........Blitzl
ichtberechnungen........";
1990 print
2000 print"......Was soll berechne
t werden?":print
2010 print
2020 print" (1) Blitzentfernung":pr
int
2030 printc4$" (2) Blitzentfernung
bei Nahaufnahmen":print
2040 printc4$" (3) Blendenzahl":pri
nt
2050 printc4$" (4) Menue"
2060 print
2070 char1,13,20,bw$
2080 geta$:ifa$=""then2080
2090 ifa$="1"then2140
2100 ifa$="2"then2230
2110 ifa$="3"then2530
2120 ifa$="4"then220
2130 goto2080
2140 printcl$q3$" Leitzahl d.Blitze
s":print:inputle
2150 print
2160 printc4$" Blende":print:inputb
l
2170 be=le/bl
2180 printc4$" Entfernung Blitz >
Motiv ="be"m"
2190 char1,13,20,wd$
2200 geta$:ifa$=""then2200
2210 ifa$="j"then2140
2220 ifa$="n"then1980
2230 printcl$:print:printc4$"..Was
soll errechnet werden?"
2240 print:print:print:printc4$"   "
;:printrn$"E"rf$"ntfernung"
2250 printc4$"   ";:printrn$"B"rf$"l
ende"
2260 char1,13,20,bw$
2270 geta$:ifa$=""then2270
2280 ifa$="e"then2410
2290 ifa$="b"then2310
2300 goto2270
2310 printcl$:printc4$" Entfernung"
:print:inputbe
2320 printc4$" Leitzahl":print:inpu
tle
2330 printc4$" Abbildungsmassst.":p
rint:inputab
2340 bl=le/be/(ab+1):bl=int((bl+.05
)*10)/10
2350 printc4$" Blende ="bl:print
2360 char1,13,20,wd$
2370 geta$:ifa$=""then2370
2380 ifa$="j"then2310
2390 ifa$="n"then1980
2400 goto2370
2410 printcl$q3$" Leitzahl d.Blitze
s":print:inputle
2420 print
2430 printc4$" Blende":print:inputb
l
2440 printc4$" Abbildungsmasst.":pr
int:inputab
2450 be=le/(bl*(ab+1))
2460 be=int((be+0.005)*100)/100
2470 printc4$" Entfernung Blitz >
Motiv ="be"m"
2480 char1,15,20,wd$
2490 geta$:ifa$=""then2490
2500 ifa$="j"then2230
2510 ifa$="n"then1980
2520 goto2490
2530 printcl$q3$" Blitzentfernung":
print:inputbe
2540 print
2550 printc4$" Leitzahl":print:inpu
tle
2560 print
2570 bl=le/be
2580 printc4$" Blende =";bl
2590 char1,13,20,wd$
2600 geta$:ifa$=""then2600
2610 ifa$="j"then2530
2620 ifa$="n"then1980
2630 goto2600
2640 rem*farbtemperatur*
2650 printcl$q3$"............Farbt
emperatur"
2660 printq3$" Berechnung der Farbt
emperatur nach Ein- ";
```

```
2670 printc4$"..gabe von 2 Werten.W
enn kein Wert vor-"
2680 printc4$".........handen,dann
0 eingeben."
2700 fort=1to2000:next
2710 printcl$q3$" Farbtemperatur de
r Beleuchtung.":print:inputa$:a=val
(a$)
2720 ifa<0then2710
2730 printc4$c4$" Farbtemperatur de
s  Filmes":print:inputb$:b=val(b$)
2740 ifb<0then2730
2750 ifa+b=0then2710
2760 printc4$"....Korrektureigensch
aft des Filters in";:print c4$"...M
ired"
2770 print:inputk$:k=val(k$)
2780 ifk<>0then2800
2790 ifa*b=0then2760
2800 c=1000000
2810 ifa<>0then2850
2820 a=c*b/(c+k*b)
2830 a=int(a+.5)
2840 goto2910
2850 ifb<>0then2890
2860 b=c*a/(c-k*a)
2870 b=int(b+.5)
2880 goto2910
2890 k=c*(b-a)/(a*b)
2900 k=int(k+.5)
2910 printcl$q3$" Ausgangs Farbtemp
eratur:";a
2920 printc4$" Filmfarbtemperatur:.
....";b
2930 printc4$" Filterwert........"
;k" Mired"
2940 char1,3,15,"Filterwert mit -
=roetl. Filter(KR)"
2950 char1,3,17,"Filterwert mit +
=blauer Filter(KB)"
2960 char1,15,20,wd$
2970 geta$:ifa$=""then2970
2980 ifa$="j"then2710
2990 ifa$="n"then220
3000 goto2970
3010 printcl$c4$qb$;
3020 print"*.......................
...............*";
3030 print"*...............Tabellen
...............*";
3040 print"*.......................
...............*";
3050 printqb$;
3060 printq3$
3070 print"...........(1) Filterda
tei...."
3080 printq3$
3090 print"...........(2) Lichttab
elle..."
3100 geta$:ifa$=""then3100
3110 ifa$="1"then3140
3120 ifa$="2"then3570
3130 goto3100
3140 printcl$c4$qb$;
3150 print"*.......................
...............*";
3160 print"*.............Filterdat
ei............*";
3170 print"*.......................
...............*";
3180 printqb$;
3190 printq3$" Sie koennen irgend e
inen Wert eingeben"
3200 printc4$"..........Sei es der
Miredwert"
3210 printc4$"..................ode
r"
3220 printc4$"..der Korrekturfaktor
 (in KB oder KR)"
3230 char1,10,20,wt$
3240 fort=1to2000:next
3250 goto3310
3260 char1,13,20,wd$
3270 getx$:ifx$=""then3270
3280 ifx$="j"then3310
3290 ifx$="n"then220
3300 goto3270
3310 printcl$
3320 line=3440:restore:print:char1,
10,5," Suchbegr.":print:inputk$
3330 l=len(k$)
3340 printcl$c4$
3350 reada$:line=line+10
3360 ifa$="§"thenprint:goto3260
3370 fori=1tolen(a$)
3380 ifmid$(a$,i,l)=k$then3410
3390 nexti
3400 goto3350
3410 char1,10,5,a$:char1,10,20,wt$
3420 fort=1to2000:goto3340
3430 data"  KR  2   + 18 Mired"
3440 data"  KR  2.5 + 27 Mired"
3450 data"  KR  3   + 35 Mired"
3460 data"  KR  6   + 50 Mired"
3470 data"  KR  9   + 81 Mired"
3480 data"  KR 12   +112 Mired"
3490 data"  KR 15   +131 Mired"
3500 data"  KB  2   - 18 Mired"
3510 data"  KB  3   - 32 Mired"
3520 data"  KB  6   - 45 Mired"
3530 data"  KB  9   - 81 Mired"
3540 data"  KB 12   -112 Mired"
3550 data"  KB 15   -131 Mired"
3560 data§
3570 printcl$;
3580 print
```

```
3590 print
3600 print"..............Farbtempera
tur...."
3610 print
3620 print"..............verschieden
er....."
3630 print
3640 print"..............Lichtquell
en....."
3650 print
3660 print
3670 print
3680 print"..........Angaben erfolg
en in"
3690 print
3700 print"............"rn$"K"rf$"e
lvin und "rn$"M"rf$"ired      "
3710 print
3720 print".............ausserdem d
er....."
3730 print
3740 print"...........Filtertyp in
KR+KB.."
3750 print
3760 print"..Alle Angaben beziehen
sich auf Tages-"
3770 print
3780 print"................lichtfil
m"
3790 print
3800 char1,10,22,wt$
3810 fort=1to2000:next
3820 printcl$;
3830 print
3840 print" Kerze................"
3850 print
3860 print" 1900Kelvin...526Mired..
Filt.= 2*KB15"
3870 print
3880 print" Lampe..40Watt........"
3890 print
3900 print" 2600Kelvin...385Mired..
Filt.= KB15+KB9"
3910 print
3920 print" Lampe..75Watt........"
3930 print
3940 print" 2800Kelvin...357Mired..
Filt.= KB15+KB6"
3950 print
3960 print" Lampe 100Watt........"
3970 print
3980 print" 2900Kelvin...345Mired..
Filt.= KB15+KB2"
3990 print
4000 print" Projektor Lampe Hochvol
t"
4010 print
4020 print" 3000Kelvin...333Mired..
Filt.= KB15+KB2"
4030 print
4040 print
4050 print"..................>>.."
4060 fort=1to2000:next
4070 printcl$;
4080 print
4090 print" Projektor Lampe Halogen
"
4100 print
4110 print" 3400Kelvin...294Mired..
Filt.= KB12"
4120 print
4130 print" Halogenlampe Hartglas"
4140 print
4150 print" 3200Kelvin...313Mired..
Filt.= KB15"
4160 print
4170 print" Halogenlampe Quarzkolbe
n "
4180 print
4190 print" 3400Kelvin...294Mired..
Filt.= KB12"
4200 print
4210 print" Nitraphotlampe..500W/27
Im "
4220 print
4230 print" 3200Kelvin...313Mired..
Filt.= KB15"
4240 print
4250 print" Nitraphotlampe..500W/34
Im "
4260 print
4270 print" 3400Kelvin...294Mired..
Filt.= KB12"
4280 print
4290 print
4300 print"..................>>.."
4310 fort=1to2000:next
4320 printcl$;
4330 print
4340 print" Blitzlampen weiss...."
4350 print
4360 print" 3800Kelvin...263Mired..
Filt.= KB9"
4370 print
4380 print" Mondlicht............"
4390 print
4400 print" 4000Kelvin...250Mired..
Filt.= KB9"
4410 print
4420 print" Morgen/Abend........."
4430 print
4440 print" 4700Kelvin...213Mired..
Filt.= KB3"
4450 print
4460 print" Elektronenblitz......"
4470 print
```

```
4480 print" 5500Kelvin...182Mired..
Filt.= 0"
4490 print
4500 print" Sonne (10-14Uhr)....."
4510 print
4520 print" 5500Kelvin...182Mired..
Filt.= 0"
4530 print
4540 print
4550 print"..................>>.."
4560 fort=1to2000:next
4570 printcl$;
4580 print
4590 print" Blitzlampen blau....."
4600 print
4610 print" 5500Kelvin...182Mired..
Filt.= 0 "
4620 print
4630 print" Leicht nebliges Wetter"
4640 print
4650 print" 8400Kelvin...119Mired..
Filt.= KA6"
4660 print
4670 print" Licht vom blauen Himmel
 (im Schatten)"
4680 print
4690 print"12000Kelvin....83Mired..
Filt.= KA12"
4700 print
4710 print" Licht vom blauen Himmel
 im Gebirge"
4720 print
4730 print" (im Schatten)........"
4740 print
4750 print"18000Kelvin....56Mired..
Filt.= KA15"
4760 print
4770 print
4780 print
4790 print"..................>>.."
4800 fort=1to2000:goto220
4810 rem foto
==================128
4830 rem 013372 bytes programm
====
4840 rem 000105 bytes variablen
====
4850 rem 000000 bytes strings
====
4860 rem 000000 bytes arrays
====
4870 rem 046364 bytes free(0)
====
4890 rem
===========================
```

```
10 rem verben englisch ========c16
20 rem (p)        commodore welt  6
30 rem ============================
40 rem (c)                        =
50 rem h-j turban                 =
60 rem                            =
70 rem version 3.5  40z/ascii     =
80 rem c16/c116/plus4 + 1531/1541=
90 rem ============================

100 c4$=chr$(17):c3$=chr$(29)
110 rn$=chr$(18):rf$=chr$(146)
120 fa$=chr$(130):fb$=chr$(131)
130 he$=chr$(19)
140 forq=1to5:q5$=q5$+c4$:nextq
150 forq=1to3:q3$=q3$+c3$:nextq
160 vol 8
170 scnclr
180 for x= 1 to 23
190 read m$
200 char 1,x,x,m$
210 next
220 for w= 1 to 4
230 for s= 1 to 3
240 for x=1 to 3
250 sound 1,596,5:sound 2,516,5
260 sound 1,596,5:sound 2,516,5
270 sound 1,572,10:sound 2,572,10
280 next x
290 sound 1,897,25 :sound2,864,25
300 next s
310 sound 1,118,25:sound2,383,25
320 next w
330 print chr$(142)
340 print chr$(27)+chr$(82)
350 rem flash und reverse mit contr
ol
360 printrn$fa$"anleitung"rf$chr$(1
31):print
370 for m=1to1400:next
380 print"es wird ihnen ein deutsch
es"
390 print:print"verb gegeben."
400 print:print"sie muessen nun der
 reihe nach"
410 print:print"die stammformen ein
geben."
420 print:print"nach jeder stammfor
m druecken"
430 print:print"sie die return tast
e."
440 print:print"beim dritten mal er
scheint"
450 print:print"die musterloesung."
460 print:print"um ein neues stichw
ort zu bekommen"
470 print:print"druecken sie eine t
aste."
```

```
480 print:print"jetzt auch!!!"
490 getkeym$
500 trap 2560
510 vol8
520 print chr$(27)+chr$(82)
530 print"S"
540 m$="-":for x=1 to 6
550 m$=m$+m$:next
560 printrf$fa$"unregelmaessige ver
ben":printchr$(131)
570 print:print"wenn sie stoppen wo
llen
580 print"dann druecken sie run/sto
p"
590 printhe$:rem home
600 printq5$q3$
610 printchr$(27)+chr$(84)
620 print chr$(14)
630 read a$,b$,c$,d$
640 printd$
650 input a1$:sound 1,400,25
660 input b1$:sound 1,400,25
670 input c1$:sound 1,400,25
680 if c1$=c$ then p=p+1
690 if a1$=a$ then p=p+1
700 if b1$=b$ then p=p+1
710 sound 1,500,25
720 print a$,b$,c$,d$
730 getkeyd$
740 scnclr
750 goto 2670
760 char1,25,2,"note:"
770 char1,32,2,n$
780 char1,25,4,"punkte:"
790 char1,32,4,p$
800 goto 600
810 data,,u,n,r,e,g,e,l,m,a,e,s,s,i
,g,
820 data v,e,r,b,e,n
830 rem verben
840 data arise,arose,arisen,erheben
850 data awake,awoke,awaked,erwache
n
860 data backslide,backslid,backsli
d,zurueckfallen
870 data bear,bore,born,tragen
880 data beat,beat,beaten,schlagen
890 data become,became,become,werde
n
900 data beget,begot,begotten,(er)z
eugen
910 data begin,began,begun,beginnen
920 data behold,beheld,beheld,erbli
cken
930 data bend,bent,bent,beugen/bieg
en
940 data bereave,bereft,bereft,bera
uben
950 data beseech,besought,besought,
anflehen
960 data bid,bade,bidden,befehlen
970 data bid,bid,bid,bieten
980 data bind,bound,bound,verbinden
990 data bite,bit,bitten,beissen
1000 data bleed,bled,bled,bluten
1010 data blow,blew,blown,blasen
1020 data break,broke,broken,breche
n
1030 data breed,bred,bred,"sich ver
mehren"
1040 data bring,brought,brought,bri
ngen
1050 data broadcast,broadcast,broad
cast,senden/tv
1060 data browbeat,browbeat,browbea
ten,einschuechtern
1070 data build,built,built,erbauen
1080 data burn,burnt,burnt,brennen
1090 data burst,burst,burst,explodi
eren
1100 data buy,bought,bought,kaufen
1110 data can,could,,koennen
1120 data cast,cast,cast,abwerfen
1130 data catch,caught,caught,fange
n
1140 data chide,chid,chidden,tadeln
1150 data choose,chose,chosen,waehl
en
1160 data cleave,clove,cloven,spalt
en
1170 data cleave,clave,,festhalten
1180 data cling,clung,clung,kleben
1190 data cloth,clothed,clothed,kle
iden
1200 data come,came,come,kommmen
1210 data cost,cost,cost,kosten
1220 data creep,crept,crept,krieche
n
1230 data cut,cut,cut,schneiden
1240 data dare,dared,dared,wagen
1250 data deal,dealt,dealt,geschaef
te-machen
1260 data dive,dived,dived,springen
/tauchen-nach
1270 data do,did,done,tun
1280 data draw,drew,drawn,zeichnen
1290 data dream,dreamt,dreamt,traum
en
1300 data drink,drank,drunk,drinken
1310 data drive,drove,driven,fahren
1320 data dwell,dwelt,dwelt,wohnen
1330 data eat,ate,eaten,essen
1340 data fall,fell,fallen,fallen
1350 data feed,fed,fed,fuettern
1360 data feel,felt,felt,fuehlen
1370 data fight,fought,fought,kaemp
```

# ANZEIGENSERVICE

**Die große Börse für jeden Zweck in der CBM REVUE / COMMODORE-WELT. Kostenlos für Privat-Inserenten. Spottbillig für gewerbliche Anbieter. Einfach Coupon ausschneiden, fotokopieren o.ä., ausfüllen und ab die Post – Freimachen nicht vergessen! – Unsere Adresse steht auf dem Coupon, ebenso die Preise für gewerbliche Anbieter! Achtung! Wir weisen ausdrücklich darauf hin, daß wir offensichtlich gewerbliche Anzeigen nicht kostenlos veröffentlichen und uns jedweden Abdruck kostenloser Anzeigen vorbehalten müssen, insbesondere, wenn deren Inhalt nicht CBM-typisch ist oder gegen geltendes Recht verstößt. Private Chiffreanzeigen werden nicht aufgenommen. Für Privatanbieter: maximal acht Zeilen à 28 Anschläge. Für gewerbliche Anbieter: 5 DM p. mm.**

**COMMODORE-WELT**
**Anzeigenabteilg.**
**Postfach 1107**
**8044 Lohhof**

Name ____________________

Vorname ____________________

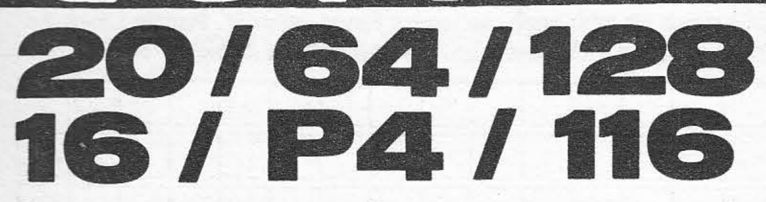
Straße/Hausnr. ____________________

Plz/Ort ____________________

✂ - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

# DAS SONDERANGEBOT: PRIVATE KLEINANZEIGEN KOSTENLOS!

Das bietet Ihnen COMMODORE-WELT: KLEINANZEIGEN SIND KOSTENLOSE FÜR PRIVATANBIETER! Suchen Sie etwas, haben Sie etwas zu verkaufen, zu tauschen, wollen Sie einen Club gründen? Coupon ausfüllen, auf Postkarte kleben oder in Briefumschlag stecken und abschicken. So einfach geht das: Wollen Sie das Heft nicht zerschneiden, können Sie den Coupon auch fotokopieren. Oder einfach den Anzeigentext uns so schicken, auf Postkarte oder im Brief. Aber bitte mit Druckbuchstaben oder in Schreibmaschinenschrift!
Und: Einschließlich Ihrer Adresse und/oder Telefonnummer sollten acht Zeilen à 28 Anschläge nicht überschritten werden.
ACHTUNG: WICHTIGER HINWEIS!
Wir veröffentlichen nur Kleinanzeigen privater Inserenten, keine gewerblichen Anzeigen. Die kosten pro Millimeter DM 5.00 plus Mehrwertsteuer!
Wir versenden für Privat-Inserenten keine Beleg-Exemplare!
Chiffre-Anzeigen sind nicht gestattet! Wir behalten uns vor, Anzeigen, die gegen rechtliche, sittliche oder sonstige Gebote verstoßen, abzulehnen!
Anzeigenabdruck in der Reihenfolge ihres Eingangs, kein Rechtsanspruch auf den Abdruck in der nächsten Ausgabe!
Die Insertion ist nicht vom Kauf des Heftes abhängig!
Wir behalten uns vor, Anzeigen, die nicht zum Themenkreis des Heftes – Computer – gehören, nicht abzudrucken oder sie nur insoweit zu berücksichtigen, wie es der Umfang des kostenlosen Anzeigenteils zuläßt.




Das unabhängige Commodore-Magazin

# KOMMT REGELMÄSSIG ZU IHNEN INS HAUS

Finden Sie Ihre COMMODORE-WELT nicht am Kiosk? Weil sie schon ausverkauft ist? Oder „Ihr" Kiosk nicht beliefert wurde? Kein Problem! Für ganze 60 DM liefern wir Ihnen per Post zwölf Hefte ins Haus (Ausland 80 DM). Einfach den Bestellschein ausschneiden – fotokopieren oder abschreiben, in einen Briefumschlag und ab per Post (Achtung: Porto nicht vergessen). COMMODORE-WELT kommt dann pünktlich ins Haus.



**WICHTIGE RECHTLICHE GARANTIE!**
Sie können diesen Abo-Auftrag binnen einer Woche nach Eingang der Abo-Bestätigung durch den Verlag widerrufen – Postkarte genügt. Ansonsten läuft dieser Auftrag jeweils für zwölf Ausgaben, wenn ihm nicht vier Wochen vor Ablauf widersprochen wird, weiter.

## NUTZEN SIE UNSEREN BEQUEMEN POSTSERVICE

# ABO-SERVICE-KARTE

(16er) **COUPON**

Ja, ich möchte von Ihrem Angebot Gebrauch machen.
Bitte senden Sie mir bis auf Widerruf ab sofort jeweils die nächsten zwölf Ausgaben an untenstehende Anschrift. Wenn ich nicht vier Wochen vor Ablauf kündige, läuft diese Abmachung automatisch weiter.

Ich nehme zur Kenntnis, daß die Belieferung erst beginnt, wenn die Abo-Gebühr dem Verlag zugegangen ist.

Name ______________________ (16)

Vorname ______________________

Straße/Hausnr. ______________________

Plz/Ort ______________________

Ich bezahle:

☐ per beiliegendem Verrechnungsscheck
☐ gegen Rechnung
☐ bargeldlos per Bankeinzug von meinem Konto

bei (Bank) und Ort ______________________

Kontonummer ______________________

Bankleitzahl ______________________
(steht auf jedem Kontoauszug)

Unterschrift ______________________

Von meinem Widerspruchsrecht habe ich Kenntnis genommen.

Unterschrift ______________________

**COMMODORE-WELT**
Abo-Service (16er)
Postfach 1107
8044 Unterschleißheim

# PROGRAMMSERVICE

Hiermit bestelle ich in Kenntnis Ihrer Verkaufsbedingungen die Listings dieses Heftes auf

☐ Kassetten zu DM 40,– ☐ Disketten zu DM 40,– (16er)

Ich zahle:
Zutreffendes bitte ankreuzen!
per beigefügtem Scheck ( ) Schein ( )

Gegen Bankabbuchung am Versandtag ( )

Meine Bank (mit Ortsname) ____________________

Meine Kontonummer ____________________

Meine Bankleitzahl ____________________ (steht auf jedem Bankauszug) ____________________

Vorname ____________________ Nachname ____________________

Str./Nr. ____________________ Plz/Ort ____________________

Verkaufsbedingungen: Lieferung nur gegen Vorkasse oder Bankabbuchung. Keine Nachnahme. Umtausch bei Nichtfunktionieren.

Unterschrift ____________________

Bitte ausschneiden und einsenden an

**COMMODORE-WELT**
KASSETTENSERVICE (16er)
Postfach 1107
8044 Unterschleißheim

---

# LESER WERBEN LESER

GEWINNEN SIE EINE COMPUTER-UHR! Und zusätzlich eventuell noch ein großes Commodore-Buch. Oder ein Paket Disketten. ODER AUCH EINEN COMMODORE-DRUCKER – ODER EINE DISKETTENSTATION! Wie? Sie werben einen Abonnenten. Dann haben Sie auf jeden Fall schon die Computer-Uhr gewonnen. Zusätzlich verlosen wir unter allen, die mitmachen, jeden Monat vier weitere wertvolle Preise. Und alle sechs Monate gibt es einen Hauptpreis unter allen Abo-Werbern zu gewinnen. Also: Mitmachen. Mitgewinnen.

Ja, ich mache mit beim Abo-Wettbewerb. Ich habe



Herrn/Frau ____________________

____________________

Straße/Hausnr. ____________________

____________________

Plz/Ort ____________________

____________________

als neuen Abonnenten für die COMMODORE WELT geworben.

Der neue Abonnent war bisher noch nicht Bezieher dieser Zeitschrift.
Als Prämie erhalte ich nach Eingang des Abo-Entgeltes auf jeden Fall eine Computer-Uhr, wie abgebildet, und nehme zusätzlich noch an der Verlosung des Monats sowie der halbjährlichen Hauptpreise teil. Mir ist bekannt, daß der Rechtsweg bei den Verlosungen ausgeschlossen ist.
Meinen Preis senden Sie an

Name ____________________

Straße/Hsnr. ____________________

Plz/Ort ____________________

(Bitte ausschneiden und zusammen mit der Abo-Bestellkarte links einsenden!) (16)

# VERDIENEN SIE GELD MIT IHREM COMPUTER!

Haben Sie einen Commodore VC 20 oder C 64? Einen 16/116, Plus 4? Oder einen 128? Können Sie programmieren? In Basic oder Maschinensprache? Dann bietet COMMODORE-WELT Ihnen die Möglichkeit, mit diesem Hobby Geld zu verdienen!

Wie? Ganz einfach. Sie senden uns die Programme, die Sie für einen Abdruck als geeignet halten, zusammen mit einer Kurzbeschreibung, aus der auch die verwendete Hardware – eventuelle Erweiterungen – benutzte Peripherie – hervorgehen muß (Schauen Sie sich dazu den Kopf unserer Programmlistings an.)

Benötigt werden: Zwei Listings des Programms sowie eine Datenkassette oder Diskette! Wenn die Redaktion sich überzeugt hat, daß dieses Programm läuft und sich zum Abdruck eignet, zahlen wir Ihnen pro Programm je nach Umfang bis zu DM 300,–!

Sollten Sie keinen Drucker haben, genügt der Datenträger.

Sie erhalten Ihre Kassette/Diskette selbstverständlich zurück, wenn Sie einen ausreichend frankierten Rückumschlag mit Ihrer Adresse beifügen.

Bei der Einsendung müssen Sie mit Ihrer Unterschrift garantieren, daß Sie der alleinige Inhaber der Urheber-Rechte sind! Benutzen Sie bitte anhängendes Formular! (Wir weisen darauf hin, daß auch die Redaktion amerikanische und englische Fachzeitschriften liest und „umgestaltete" Programme ziemlich schnell erkennt).

Um Ihnen die Arbeit zu erleichtern, finden Sie hier ein Formular. Sie können es ausschneiden oder fotokopieren.

Name des Einsenders: ______________________
Straße/Hausnr./Tel.: ______________________
Plz/Ort: ______________________

Hiermit biete ich Ihnen zum Abdruck folgende(s) Programm(e) an:

______________________
______________________
______________________
______________________

Benötigte Geräte: ______________________
______________________

Beigefügt ( ) Listings ( ) Kassette ( ) Diskette

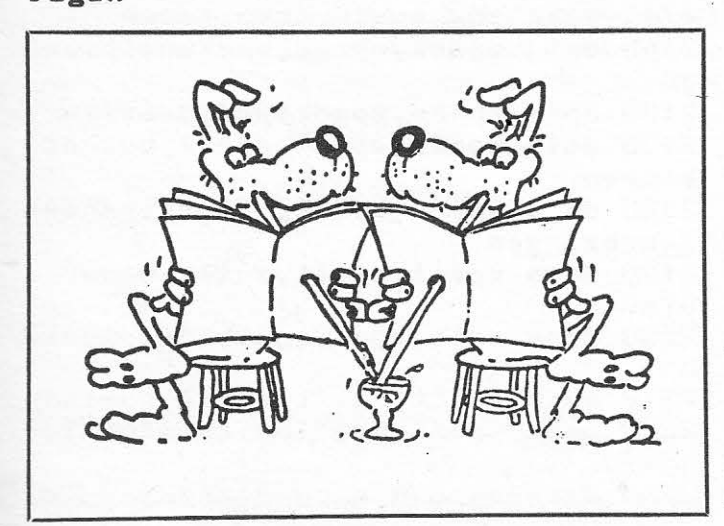
Ich versichere, der alleinige Urheber des Programmes zu sein!
Hiermit ermächtige ich die Redaktion, dieses Programm abzudrucken und wirtschaftlich zu verwerten. Sollte es in den Kassetten-Service aufgenommen werden, erhalte ich auch dafür eine entsprechende Vergütung, das Copyright geht insoweit auf den Verlag über.

______________________
Rechtsverbindliche Unterschrift

**COMMODORE-WELT**
Postfach 1107
8044 Lohhof

```
fen
1380 data find,found,found,finden
1390 data flee,fled,fled,fliehen
1400 data fling,flung,flung,schleud
ern
1410 data fly,flew,flown,fliegen
1420 data forbear,forbore,forborne,
unterlassen
1430 data forbid,forbade,forbidden,
verbieten
1440 data forget,forgot,forgotten,v
ergessen
1450 data forgive,forgave,forgiven,
vergeben
1460 data forsake,forsook,forsaken,
verlassen
1470 data swear,swore,sworne,schwoe
ren
1480 data freeze,froze,frozen,frier
en
1490 data get,got,got,bekommen
1500 data give,gave,given,geben
1510 data hang,hung,hung,haengen
1520 data hang,hanged,hanged,hencke
n
1530 data have,hase,had,haben
1540 data hear,heard,heard,hoeren
1550 data heave,heaved,heaved,hiefe
n
1560 data go,went,gone,gehen
1570 data hew,hewed,hewn,hacken
1580 data hide,hid,hidden,verstecke
n
1590 data hit,hit,hit,schlagen/tref
fen
1600 data hold,held,held,halten
1610 data hurt,hurt,hurt,verletzen
1620 data keep,kept,kept,behalten
1630 data kneel,knelt,knelt,knien
1640 data knit,knitted,knitted,stri
cken
1650 data know,knew,known,kennen
1660 data lade,laded,laded,beladen
1670 data lay,laid,laid,legen
1680 data lead,led,led,fuehren
1690 data lean,leant,leant,neigen/b
iegen
1700 data leap,leapet,leapet,spring
en
1710 data leave,left,left,verlassen
1720 data lend,lent,lent,verleihen
1730 data let,let,let,lassen
1740 data lie,lay,lain,liegen
1750 data lose,lost,lost,verlieren
1760 data make,made,made,machen
1770 data may,might,,koennte
1780 data mean,meant,meant,meinen
1790 data mow,mowed,mown,maehen
1800 data pay,paid,paid,zahlen
1810 data put,put,put,stelen/legen
1820 data read,read,read,lesen
1830 data rend,rent,rent,zerreissen
1840 data ride,rode,ridden,reiten
1850 data ring,rang,rung,klingeln
1860 data rise,rose,risen,aufstehen
/gehen
1870 data rive,rived,riven,reissen
1880 data roast,roasted,roasted,roe
sten
1890 data run,ran,run,laufen
1900 data saw,sawed,sawn,saehen
1910 data say,said,said,sagen
1920 data see,saw,seen,sehen
1930 data seek,sought,sought,sehen
1940 data seethe,seethed,seethed,si
eden
1950 data sell,sold,sold,verkaufen
1960 data send,sent,sent,schicken
1970 data set,set,set,stellen
1980 data sew,sewed,sewn,naehen
1990 data shake,shook,shaken,schuet
teln
2000 data shall,should,,sollen
2010 data shear,sheared,sheared,(sc
haf)scheren
2020 data shed,shed,shed,vergiessen
2030 data shine,shone,shone,scheine
n
2040 data shoot,shot,shot,schiessen
2050 data show,showed,shown,zeigen
2060 data shut,shut,shut,schliessen
2070 data sing,sang,sung,singen
2080 data sink,sank,sunk,sinken
2090 data sit,sat,sat,setzen
2100 data slay,slew,slain,erschlage
n
2110 data sleep,slept,slept,schlafe
n
2120 data slide,slit,slit,rutschen
2130 data smell,smelt,smelt,riechen
2140 data sow,sowed,sown,saeen
2150 data speak,spoke,spoken,sprech
en
2160 data speed,sped,sped,beeilen
2170 data spell,spelt,spelt,buchsta
bieren
2180 data spend,spent,spent,spenden
/verbringen
2190 data spill,spilt,spilt,ueberla
ufen
2200 data spit,spat,spat,aufspiesse
n
2210 data split,split,split,spalten
2220 data spoil,spoiled,spoilt,plue
ndern
2230 data spread,spread,spread,aus/
```

```
verbreiten
2240 data spring,sprang,sprung,spri
ngen
2250 data stand,stood,stood,stehen
2260 data steal,stole,stolen,stehle
n
2270 data stick,stuck,stuck,stecken
/stechen
2280 data sting,stung,stung,aufstac
heln
2290 data stink,stank,stunk,stinken
2300 data strew,strewed,strewn,stre
uen
2310 data strike,struk,struk,streik
en
2320 data string,strung,strung,aufr
eihen
2330 data strive,strove,striven,"si
ch bemuehen"
2340 data swear,swore,sworn,schwoer
en
2350 data sweep,swept,swept,kehren/
fegen
2360 data swim,swam,swum,schwimmen
2370 data swing,swung,swung,schwing
en
2380 data take,took,taken,nehmen
2390 data teach,taught,taught,lehre
n
2400 data tell,told,told,erzaehlen
2410 data think,thought,thought,den
ken
2420 data thrive,throve,thriven,ged
eihen
2430 data throw,threw,thrown,werfen
2440 data thrust,thrust,thrust,stos
sen
2450 data tread,trod,trodden,(auf)t
reten
2460 data wake,woke,woken,aufwecken
2470 data wear,wore,worn,-kleider-t
ragen
2480 data weave,wove,woven,(ver)web
en
2490 data weep,wept,wept,(be)weinen
2500 data will,would,,werden
2510 data win,won,won,gewinnen
2520 data  ork,worked,worked,arbeit
en
2530 data wring,wrung,wrung,wringen
2540 data write,wrote,written,schre
iben
2550 rem ende der verben
2560 print chr$(14)+chr$(82)
2570 print"sie haben von 685 punkte
n"
2580 print
2590 printp,"punkte geschaft":print
2600 if p=0 then n=6:goto 2640
2610 n=int(685/(p*1.3))
2620 if n>6then n=6
2630 if n<1 then n=1
2640 print"das entspricht der note"
2650 print:printspc(5)n
2660 end
2670 if p=0then n=6:goto 2710
2680 n=int(685/(p*1.3))
2690 if n>6then n=6
2700 if n<1 then n=1
2710 p$=str$(p)
2720 n$=str$(n)
2730 goto 760
2740 rem unregelmaessige verben
====c16
2750 rem 12777 bytes memory
===
2760 rem 08176 bytes programm
===
2770 rem 04601 bytes free
===
2780 rem
=============================
```



# TELE-MÜHLE

Ich habe mich um eine originalgetreue Simulation des Spielklassikers „Mühle" bemüht und alle graphischen Möglichkeiten des V3.5- Basics weidlich ausgenutzt. Endlich ein Gesellschaftsspiel auf dem Terminal, durch dessen User-freundliche Handhabung die beiden Spieler schnell das Original vergessen haben werden!

Denn bei Screenwechseln (z.B. Steinwegnahme) wird nicht, wie bei ähnlichen Spielen üblich, der ganze Bildschirm gelöscht und verändert neu wieder aufgebaut, sondern wirklich nur die betreffende Stelle verändert. Selbstverständlich (und leider) läuft ein Programm solchen Umfangs nicht auf der Grundversion des C 16/116, da die der Grafik verbleibenden 2 K-Bytes wirklich zu wenig sind. Aber allen Plus 4 - und C16/116 + Ausbau-

modul-Besitzern, die ja schon länger auf dem Markt zu erhalten sind (16K oder 64K), steht dieses schöne Programm zur Verfügung.

Beim Spielen selbst hat man durch die einfache Handhabung keinerlei Schwierigkeiten. Selbstverständlich beherrscht der Computer sämtliche Regeln des Mühlespieles und macht sich bei einem Regelverstoß lautstark bemerkbar und ruft den Sünder zur Ordnung.

Nur bei einer Regel (bei Steinentnahme - nur möglich, wenn der Stein nicht in einer Mühle ist, falls nicht alle Steine in Mühlen sind), fragt der Computer den betroffenen Spieler, ob alles seine Richtigkeit hat.

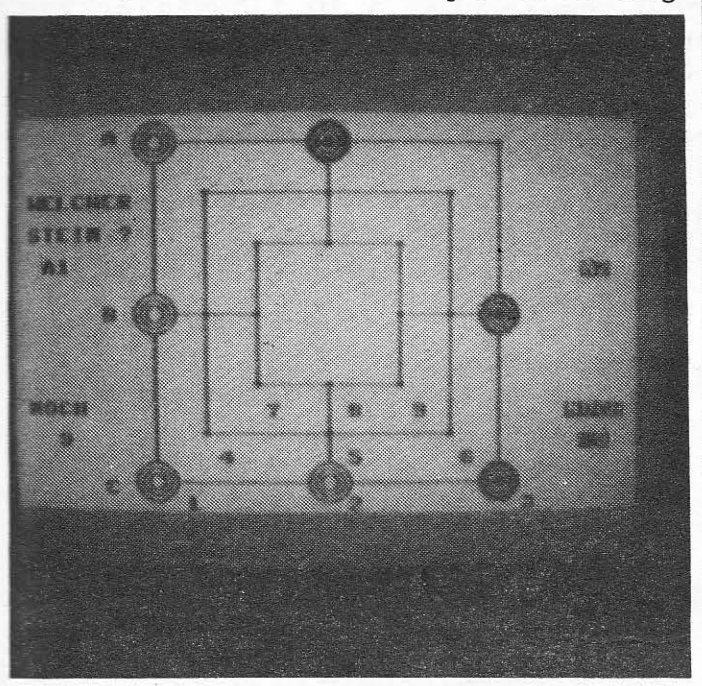
Sämtliche übrigen Regeln beherrscht der Computer aber perfekt (z.B. Mühle-Erkennung oder Hüpfen bei drei Steinen, usw.) Hat nun ein Spieler gesiegt, auf welche Weise auch immer, wird er am Ende mit einer Fanfare und schönen grafischen Spielereien belohnt.

## Zum Spielfeld

Während des Spielverlaufs sieht man ganz oben links für Weiß und rechts für Schwarz die Position des letzten weggenommenen Steines, darunter die letzte eingetippte Steinposition und ganz unten die jeweilige Steinanzahl. Das Spielfeld ist waagrecht dreigeteilt, in A-B-C, und senkrecht bestimmen jeweils Zahlen die genauen Positionen. Sieht man sich das Einteilungssystem genauer an, versteht man es sofort und wird keine Schwierigkeiten damit haben.

Zu beachten ist allerdings, daß in der B-Ebene nur die Zahlen 1, 4, 7, 9, 6 und 3 existieren, um das logische Durchzählsystem in den beiden anderen Ebenen nicht zu durchbrechen.

Wie üblich beginnt Weiß (Rand gelb) und wechselt mit Schwarz (Rand blau) ab. Zuerst werden alle Steine gesetzt (‚Wohin'), und dann wird gezogen (‚Von' - ‚Nach').




**Mühle per Computer**

```
10 rem tele-muehle ============p/4
20 rem (p) by    commodore welt bw
30 rem ============================
40 rem (c) by                     =
50 rem m.purucker                 =
60 rem                            =
70 rem version 3.5 40z/ascii      =
80 rem plus/4+ C 16 m. 64 K       =
90 rem ============================

100 graphic 1,1:graphic 0,1
110 rem strings initialisieren****
120 trap 3440
130 :
140 dim s$(26):color 0,2,0:color 4,
2,0
150 restore 940:for uu=0to11:read n
,p:gosub 950:next uu
160 s$(12)=chr$(98)+chr$(32)+chr$(9
8)+chr$(32)+chr$(98)+chr$(32)+chr$(
98)+chr$(32)
170 s$(12)=s$(12)+chr$(98):s$(13)=c
hr$(98)+chr$(32)+chr$(98):s$(14)=ch
r$(117)
180 s$(15)=chr$(105):s$(16)=chr$(10
6):s$(17)=chr$(107):s$(18)=chr$(98)
:s$(19)=chr$(32)
190 s$(20)=chr$(99):s$(21)=chr$(176
):s$(22)=chr$(174):s$(23)=chr$(181)
200 s$(24)=chr$(182):s$(25)=chr$(17
3):s$(26)=chr$(189)
210 :
220 rem vorspann******************
230 scnclr
240 char1,1,1,s$(14)+s$(1)+s$(15):c
har1,1,2,s$(16)+s$(10)+s$(15)+s$(14
)+s$(2)+s$(17)
250 char1,5,3,s$(0)+s$(3)+s$(14)+s$
(4)+s$(15)+s$(9)+s$(14)+s$(15)+s$(5
)+s$(14)+s$(4)
260 char1,36,3,s$(15):char1,5,4,s$(
0)+s$(3)+s$(18)+s$(14)+s$(6)+s$(17)
+s$(9)+s$(0)
270 char1,23,4,s$(5)+s$(18)+s$(14)+
s$(6)+s$(17)
280 char1,5,5,s$(0)+s$(3)+s$(0)+s$(
5)+s$(0)+s$(5)+s$(0)
290 char1,5,6,s$(0)+s$(3)+s$(18)+s$
(16)+s$(6)+s$(15)+s$(9)+s$(0)+s$(5)
+s$(18)+s$(16)
300 char1,32,6,s$(6)+s$(15)
310 char1,5,7,s$(0)+s$(3)+s$(18)+s$
(14)+s$(6)+s$(17)+s$(9)+s$(0)
320 char1,23,7,s$(5)+s$(18)+s$(14)+
s$(6)+s$(17)
330 char1,5,8,s$(0)+s$(3)+s$(0)+s$(
5)+s$(0)+s$(5)+s$(0)
340 char1,5,9,s$(0)+s$(3)+s$(18)+s$
```

```
(16)+s$(6)+s$(15)+s$(9)+s$(18)+s$(1
6)+s$(6)+s$(15)
350 char1,28,9,s$(9)+s$(18)+s$(16)+
s$(6)+s$(15)
360 char1,5,10,s$(16)+s$(17)+s$(3)+
s$(16)+s$(4)+s$(17)+s$(9)+s$(16)+s$
(4)+s$(17)
370 char1,28,10,s$(9)+s$(16)+s$(4)+
s$(17)
380 char1,2,13,s$(21)+s$(20)+s$(22)
+s$(19)+s$(21)+s$(20)+s$(22)+s$(19)
+s$(21)+s$(19)
390 char1,12,13,s$(23)+s$(24)+s$(19
)+s$(22)+s$(19)+s$(21)+s$(8)+s$(22)
+s$(19)
400 char1,24,13,s$(21)+s$(7)+s$(21)
+s$(6)+s$(22)
410 char1,2,14,s$(12)+s$(8)+s$(13)+
s$(8)+s$(13)+s$(7)+s$(18)
420 char1,2,15,s$(12)+s$(8)+s$(13)+
s$(8)+s$(13)+s$(7)+s$(18)
430 char1,2,16,s$(12)+s$(8)+s$(13)+
s$(11)+s$(13)+s$(7)+s$(18)+s$(11)
440 char1,2,17,s$(18)+s$(19)+s$(25)
+s$(20)+s$(26)+s$(19)+s$(13)+s$(8)+
s$(13)+s$(8)
450 char1,22,17,s$(13)+s$(7)+s$(18)
:char1,2,18,s$(25)+s$(3)+s$(26)+s$(
19)+s$(25)
460 char1,11,18,s$(6)+s$(26)+s$(19)
+s$(25)+s$(8)+s$(26)+s$(19)+s$(25)+
s$(6)
470 char 1,29,18,s$(26)+s$(19)+s$(2
5)+s$(6)+s$(26)
480 char 1,13,21,s$(14)+s$(20)+s$(1
5):char 1,13,22,s$(18)+"c"+s$(18)
490 char 1,17,22,"by purisoft m.p.
85/86":char 1,13,23,s$(16)+s$(20)+s
$(17)
500 clr:fa$=chr$(130):fo$=chr$(131)
:trap 3440:gb=9:gw=9
510 for ui=0to7:vol ui:sound 1,ui,4
0:sound 2,ui,40:color 0,ui+3,3:colo
r 4,ui+5,3
520 for t=1to400:next:next
530 sound 1,7,120:sound`2,516,120:c
olor 0,14,1:color 4,14,1
540 sound 1,516,120:sound 2,770,120
:color 0,12,2:color 4,12,2
550 sound 1,770,150:sound 2,897,150
:color 0,8,4:color 4,8,4
560 vol 8:sound 1,7,30:sound 2,516,
30:color 0,3,0:color 4,3,0
570 sound 2,516,100:sound 1,897,200
:color 0,16,3:color 4,16,3
580 for ui=8to1step-1:vol ui:sound3
,516,20:color0,5,2:color4,5,2:for t
=1to500:next
590 vol ui-1:sound 3,7,20:color 0,2
,0:color 4,2,0:for t=1to500:next:ne
xt
600 scnclr:char 1,1,1,"vorname spie
ler 1 (weiss) ":input n1$
610 char 1,1,3,"vorname spieler 2 (
schwarz) ":input n2$
620 char 1,1,10,"mit 'e' zum spiel"
630 char
1,3,24,"- mit 'space' zu erlaeuteru
ngen -"
640 poke 239,0:getkey hs$:if hs$="e
" then 960
650 scnclr:char 1,10,1,"spielerklae
rung"
660 char 1,1,3,"spielfeld:der spiel
plan ist horizontal"
670 char 1,1,5,"dreigeteilt (a-b-c)
;vertikal bestimmen"
680 char 1,1,7,"zahlen die position
en,indem man ein-"
690 char 1,1,9,"fach in den jeweili
gen muehleebenen"
700 char 1,1,11,"von links nach rec
hts durchzaehlt."
710 char 1,1,13,"in der b-ebene exi
stieren b2,b5,b8"
720 char 1,1,15,"nicht, damit das z
aehlsystem nicht"
730 char 1,1,17,"durchbrochen wird.
740 char 1,5,24,"- mit 'space' umbl
aettern -":poke 239,0:getkey hs$:sc
nclr
750 char 1,1,1,"regeln:es wird nach
 den allgemeinguel-"
760 char 1,1,3,"tigen regeln gespie
lt;eines ist aber "
770 char 1,1,5,"zu beachten:hat man
 bei der steinweg-"
780 char 1,1,7,"nahme einen stein g
ewaehlt, fraegt das"
790 char 1,1,9,"programm, ob dieses
 der regel gemaess"
800 char 1,1,11,"ist.diese regel la
utet:ein stein,"
810 char 1,1,13,"der in einer muehl
e eingeschlossen ist"
820 char 1,1,15,"darf nicht entnomm
en werden,":color 1,3,5:char1,30,15
,"falls"
830 color 1,1:char 1,1,17,"nicht al
le steine in muehlen ein-"
840 char 1,1,19,"geschlossen sind."
850 char 1,5,24,"- mit 'space' umbl
aettern -":poke 239,0:getkey hs$:sc
nclr
860 char 1,1,1,"vorsicht !:falls si
e beim steineziehen"
```

```
870 char 1,1,3,"auf den selben plat
z ziehen, auf dem"
880 char 1,1,5,"sie stehen (bzw. st
anden), haben sie "
890 char 1,1,7,"das spiel verloren,
 da der computer "
900 char 1,1,9,"dann annimmt, dass
sie mit keinem "
910 char 1,1,11,"stein mehr ziehen
koennen."
920 char 1,5,24,"- mit space zum sp
iel -":poke 239,0:getkey hs$
930 goto 960
940 data 2,98,36,99,31,99,5,32,5,99
,7,32,4,99,6,32,4,32,2,32,3,99,4,18
4
950 for tt=1ton:s$(uu)=s$(uu)+chr$(
p):next:return
960 rem hpr*******************
970 color 0,2:color 4,7,6
980 vol 5:gosub1060
990 gosub 1300
1000 gosub 1890
1010 gosub 2000
1020 gosub 2230
1030 if ko=1 then gosub 2930
1040 goto 1020
1050 rem graphicvorbereitung**
1060 graphic 1,1
1070 char 1,1,20,"rem graphische sp
ielfeldvorbereitungen"
1080 draw 1,100,30to100,20to110,20:
circle1,100,20,1:sshape
ol$,90,10,110,30
1090 draw 1,150,30to150,20to140,20:
circle1,150,20,1:sshape
od$,140,10,160,30
1100 draw 1,200,10to200,20to210,20:
circle1,200,20,1:sshape
ul$,190,10,210,30
1110 draw 1,250,10to250,20to240,20:
circle1,250,20,1:sshape
ur$,240,10,260,30
1120 draw 1,290,20to310,20:draw
1,300,20to300,30:circle1,300,20,1
1130 sshape mo$,290,10,310,30
1140 draw 1,10,100to20,100:draw
1,20,90to20,110:circle1,20,100,1
1150 sshape mr$,10,90,30,110
1160 draw 1,50,90to50,100:draw
1,40,100to60,100:circle1,50,100,1
1170 sshape mu$,40,90,60,110
1180 draw 1,110,100to100,100:draw
1,100,90to100,110:circle1,100,100,1
1190 sshape ml$,90,90,110,110
1200 draw 1,140,100to160,100:draw
1,150,90to150,110:circle1,150,100,1
1210 sshape k$,140,90,160,110
1220 circle 1,20,20,10:circle
1,20,20,7:circle 1,20,20,4
1230 sshape w$,10,10,30,30
1240 circle 0,20,20,7:circle 0,20,2
0,4:paint 1,20,20
1250 circle 0,20,20,7:circle 0,20,2
0,4
1260 sshape bl$,10,10,30,30
1270 circle 1,30,30,1:paint 1,30,30
:sshape p$,29,29,31,31
1280 scnclr:return
1290 rem spielfeldaufbau*******
1300 box 1,74,14,244,184
1310 box 1,99,39,219,159
1320 box 1,124,64,194,134
1330 draw 1,74,99to124,99
1340 draw 1,194,99to244,99
1350 draw 1,159,14to159,64
1360 draw 1,159,184to159,134
1370 restore1960:for t=1to24
1380 read x,y
1390 gshape p$,x-1,y-1
1400 next:return
1410 rem steinposition bestimmen
1420 restore1960:for t=1topo
1430 read x,y
1440 next:return
1450 rem muehlebestimmung*******
1460 ko=0:i=asc(b$)-64:j=val(z$):fo
re=1to3 step 2
1470 forf=1to7 step 3 :ka=0
1480 if f(e,f)=f(e,f+1)andf(e,f)=f(
e,f+2)and(f(e,f)=1orf(e,f)=2)thenka
=f(e,f):ko=1
1490 ifko=1and(e=iand(f=jor(f+1)=jo
r(f+2)=j))then ko=1:else ko=0
1500 if ko=1 then return
1510 next f,e
1520 for f=1to7 step 3:ka=0
1530 if f(1,f)=f(2,f)andf(1,f)=f(3,
f)and(f(1,f)=1orf(1,f)=2)thenka=f(1
,f):ko=1
1540 ifko=1and(f=jand(i=1ori=2ori=3
))thenko=1:elseko=0
1550 if ko=1 then return
1560 nextf
1570 for f=3to9 step 3:ka=0
1580 if f(1,f)=f(2,f)andf(1,f)=f(3,
f)and(f(1,f)=1orf(1,f)=2)thenka=f(1
,f):ko=1
1590 ifko=1and(f=jand(i=1ori=2ori=3
))thenko=1:elseko=0
1600 if ko=1 then return
1610 next f
1620 ka=0
1630 iff(2,1)=f(2,4)andf(2,1)=f(2,7
)and(f(2,1)=1orf(2,1)=2)thenka=f(2,
1):ko=1
```

```
1640 ifko=1and(i=2and(j=1orj=4orj=7
))thenko=1:elseko=0
1650 if ko=1 then return
1660 ka=0
1670 iff(2,3)=f(2,6)andf(2,3)=f(2,9
)and(f(2,3)=1orf(2,3)=2)thenka=f(2,
3):ko=1
1680 ifko=1and(i=2and(j=3orj=6orj=9
))thenko=1:elseko=0
1690 if ko=1 then return
1700 ka=0
1710 iff(1,2)=f(1,5)andf(1,2)=f(1,8
)and(f(1,2)=1orf(1,2)=2)thenka=f(1,
2):ko=1
1720 ifko=1and(i=1and(j=2orj=5orj=8
))thenko=1:elseko=0
1730 if ko=1 then return
1740 ka=0
1750 iff(3,2)=f(3,5)andf(3,2)=f(3,8
)and(f(3,2)=1orf(3,2)=2)thenka=f(3,
2):ko=1
1760 ifko=1and(i=3and(j=2orj=5orj=8
))thenko=1:elseko=0
1770 return
1780 rem ersatzshapebestimmung***
1790 ifpo=1orpo=4orpo=7thenez$=ol$:
return:rem ersatzsteinpositionsents
cheidung
1800 ifpo=3orpo=6orpo=9thenez$=od$:
return
1810 ifpo=16orpo=19orpo=22thenez$=u
l$:return
1820 ifpo=18orpo=21orpo=24thenez$=u
r$:return
1830 ifpo=2orpo=23thenez$=mo$:retur
n
1840 ifpo=12orpo=15thenez$=mr$:retu
rn
1850 ifpo=8orpo=17thenez$=mu$:retur
n
1860 ifpo=10orpo=13thenez$=ml$:retu
rn
1870 ez$=k$:return
1880 rem bezeichnungsaufbau********
1890 char 1,6,1,"a":char 1,6,12,"b"
:char 1,6,23,"c"
1900 char 1,11,24,"1":char 1,21,24,
"2":char 1,32,24,"3"
1910 char 1,13,21,"4":char 1,21,21,
"5":char 1,28,21,"6"
1920 char 1,16,18,"7":char 1,21,18,
"8":char 1,25,18,"9"
1930 char1,1,18,"noch":char1,35,18,
"noch",1
1940 char1,2,20,str$(gw):char1,36,2
0,str$(gb),1:return
1950 rem databloecke zu steinpositi
onsbestimmung
1960 data 74,14,159,14,244,14,99,39
,159,39,219,39,124,64,159,64,194,64
,74,99,99,99
1970 data 124,99,194,99,219,99,244,
99,74,184,159,184
1980 data 244,184,99,159,159,159,21
9,159,124,134,159,134,194,134
1990 rem steine setzen******
2000 gb=9:gw=9:w=9:b=9:for t=1to3:f
or y=1to9:f(t,y)=0:nexty,t
2010 ge$="a1a2a3a4a5a6a7a8a9b1b4b7b
9b6b3c1c2c3c4c5c6c7c8c9"
2020 trap 2030
2030 color 4,8,6:char 1,1,5,"wohin"
:poke 239,0:getkey b$,z$:a$=b$+z$:d
r=1
2040 char 1,2,9,a$
2050 ifa$="b2"ora$="b5"ora$="b8" th
ensound 1,0,20:goto 2030
2060 if f((asc(b$)-64),val(z$))<>0
then sound 1,0,20:goto 2030
2070 po=int((instr(ge$,a$)+1)/2):if
po<>0thengosub1420:elsesound1,0,20:
goto 2030
2080 gshapew$,x-10,y-10:w=w-1:f((as
c(b$)-64),val(z$))=1:gosub1890:gosu
b1460
2090 if ko=1 then gosub 2930
2100 if gw<3then3200
2110 char 0,1,5,"wohin"
2120 trap 2130
2130 color 4,15,1:char 1,34,5,"wohi
n",1:poke 239,0:getkey b$,z$:a$=b$+
z$:dr=2
2140 char 1,36,9,a$,1
2150 ifa$="b2"ora$="b5"ora$="b8" th
ensound 1,0,20:goto 2130
2160 if f((asc(b$)-64),val(z$))<>0
then sound 1,0,20:goto 2130
2170 po=int((instr(ge$,a$)+1)/2):if
po<>0thengosub1420:elsesound1,0,20:
goto 2130
2180 gshape bl$,x-10,y-10:b=b-1:f((
asc(b$)-64),val(z$))=2:gosub 1890:g
osub 1460
2190 if gb<3then3200
2200 if ko=1 then gosub 2930
2210 char 0,34,5,"wohin"
2220 if b=0thenreturn:else 2020
2230 rem steine ziehen****
2240 trap 2260
2250 ifgw<3 then3200
2260 color 4,8,6:char 1,1,5,"von  "
:poke 239,0:getkey b$,z$:a$=b$+z$:d
r=1
2270 if f((asc(b$)-64),val(z$))<>dr
 then sound 1,0,20:goto 2260
2280 char 1,2,9,a$
```

```
2290 f((asc(b$)-64),val(z$))=0:po=i
nt((instr(ge$,a$)+1)/2):bo=po
2300 if bo<>0 then gosub 1790:else
sound 1,0,20:goto 2260
2310 gosub 1420:gshape ez$,x-10,y-1
0:gosub 1890
2320 trap 2330
2330 char 1,1,5,"nach":getkey
b$,z$:a2$=b$+z$
2340 char 1,2,9,a2$
2350 ifgw<>3 then gosub2600:else ev
=1
2360 if f((asc(b$)-64),val(z$))<>0o
rev=0then sound 1,0,20:goto 2330
2370 f((asc(b$)-64),val(z$))=1:po=i
nt((instr(ge$,a2$)+1)/2)
2380 if po<>0 then gosub 1420:else
sound 1,0,20:goto 2330
2390 gshape w$,x-10,y-10:gosub 1890
:gosub 1460
2400 if ko=1 then gosub 2930
2410 char 0,1,5,"nach"
2420 trap 2440
2430 ifgb<3 then3200
2440 color4,15,1:char1,35,5,"von",1
:char1,4,5,"  ":poke239,0:getkeyb$,
z$:a$=b$+z$:dr=2
2450 if f((asc(b$)-64),val(z$))<>dr
 then sound 1,0,20:goto 2440
2460 char 1,36,9,a$,1
2470 f((asc(b$)-64),val(z$))=0:po=i
nt((instr(ge$,a$)+1)/2):bo=po
2480 if bo<>0 then gosub 1790:else
sound 1,0,20:goto 2440
2490 gosub 1420:gshape ez$,x-10,y-1
0:gosub 1890
2500 trap 2510
2510 char 1,34,5,"nach",1:poke 239,
0:getkey b$,z$:a2$=b$+z$
2520 char 1,36,9,a2$,1
2530 if gb<>3thengosub 2600:else ev
=1
2540 if f((asc(b$)-64),val(z$))<>0
or ev=0 then sound 1,0,20:goto 2510
2550 f((asc(b$)-64),val(z$))=2:po=i
nt((instr(ge$,a2$)+1)/2)
2560 if po<>0 then gosub 1420:else
sound 1,0,20:goto 2510
2570 gshape bl$,x-10,y-10:gosub 189
0:gosub 1460
2580 char 0,34,5,"nach"
2590 return
2600 rem zugbeschraenkung
2610 ev=0
2620 if a$=a2$ and dr=1 then dr=2:g
oto 3200
2630 if a$=a2$ and dr=2 then dr=1:g
oto 3200
2640 if bo>13 then 2670
2650 onbogosub2690,2700,2710,2720,2
730,2740,2750,2760,2770,2780,2790,2
800,2810
2660 return
2670 bo=bo-13:onbogosub2820,2830,28
40,2850,2860,2870,2880,2890,2900,29
10,2920
2680 bo=bo+13:return
2690 if a2$="a2"ora2$="b1"thenev=1:
return:else return
2700 if a2$="a1"ora2$="a3"ora2$="a5
"thenev=1:return:else return
2710 if a2$="a2"ora2$="b3"thenev=1:
return:else return
2720 if a2$="b4"ora2$="a5"thenev=1:
return:else return
2730 if a2$="a4"ora2$="a6"ora2$="a2
"ora2$="a8"thenev=1:return:else ret
urn
2740 if a2$="a5"ora2$="b6"thenev=1:
return:else return
2750 if a2$="b7"ora2$="a8"thenev=1:
return:else return
2760 if a2$="a7"ora2$="a5"ora2$="a9
"thenev=1:return:else return
2770 if a2$="a8"ora2$="b9"thenev=1:
return:else return
2780 if a2$="a1"ora2$="c1"ora2$="b4
"thenev=1:return:else return
2790 if a2$="b1"ora2$="b7"ora2$="a4
"ora2$="c4"thenev=1:return:else ret
urn
2800 if a2$="b4"ora2$="a7"ora2$="c7
"thenev=1:return:else return
2810 if a2$="b6"ora2$="a9"ora2$="c9
"thenev=1:return:else return
2820 if a2$="b9"ora2$="b3"ora2$="a6
"ora2$="c6"thenev=1:return:else ret
urn
2830 if a2$="b6"ora2$="a3"ora2$="c3
"thenev=1:return:else return
2840 if a2$="b1"ora2$="c2"thenev=1:
return:else return
2850 if a2$="c1"ora2$="c3"ora2$="c5
"thenev=1:return:else return
2860 if a2$="c2"ora2$="b3"thenev=1:
return:else return
2870 if a2$="b4"ora2$="c5"thenev=1:
return:else return
2880 if a2$="c4"ora2$="c6"ora2$="c2
"ora2$="c8"thenev=1:return:else ret
urn
2890 if a2$="c5"ora2$="b6"thenev=1:
return:else return
2900 if a2$="b7"ora2$="c8"thenev=1:
return:else return
2910 if a2$="c7"ora2$="c9"ora2$="c5
```

```
"thenev=1:return:else return
2920 if a2$="c8"ora2$="b9"thenev=1:
return:else return
2930 rem steinentfernen*****
2940 au=0
2950 sound 1,950,50:sound 2,850,50:
for t=1to200:next:trap 2970
2960 sound 1,150,50:sound 2,50,50
2970 char 1,1,5,"welcher":char 1,1,
7,"stein ?"
2980 poke 239,0:getkey b$,z$:we$=b$
+z$:po=int((instr(ge$,we$)+1)/2)
2990 if po=0 then sound 1,0,20:goto
 2970
3000 if dr=1 then char 1,36,2,we$,1
:goto 3020
3010 char 1,2,2,we$
3020 if f((asc(b$)-64),val(z$))=dr
or f((asc(b$)-64),val(z$))=0 then 2
960
3030 for t=1to1000:next
3040 gosub 3130
3050 if au=1 then 3070
3060 goto 2970
3070 gosub 1420:gosub 1790:f((asc(b
$)-64),val(z$))=0:au=0
3080 gshape ez$,x-10,y-10:char 0,1,
5,"welcher":char 0,1,7,"stein ?":go
sub 1890
3090 if dr=1 thengb=gb-1:char 1,36,
20,str$(gb),1
3100 if dr=2 then gw=gw-1:char 1,2,
20,str$(gw)
3110 return
3120 rem upr zu steinentfernung**
3130 graphic 0,1:char 1,7,5,"ist di
ese steinentnahme"
3140 char 1,7,7,"der regel gemaess
?"
3150 if dr=1 then va=2:else va=1
3160 char 1,2,9,"(von spieler"+str$
(va)+" zu beantworten)"
3170 char 1,13,11,"(j/n)"
3180 poke 239,0:getkey ri$:if ri$="
j" then au=1
3190 graphic 1,0:return
3200 rem schluss-siegerehrung****
3210 color 0,2,0:color 4,2,0:graphi
c 1,1:graphic 0,1
3220 if dr=1then n3$=n1$:else n3$=n
2$
3230 color 1,4,5:for t=0to24:char 1
,1,t,"gratuliere !":next
3240 color 1,12,4:char 1,20,12,fa$+
n3$+fo$
3250 color 1,8,5:char 1,25,4,"siege
r ! ! !":char 1,25,20,"sieger ! ! !
"
3260 color 1,3,5:vol 8:sound1,7,480
:sound2,770,480:gosub 3340
3270 sound 1,7,120:sound 2,516,120
3280 sound 1,516,120:sound 2,770,12
0
3290 sound 1,770,150:sound 2,897,15
0:fort=1to2300:next
3300 graphic 1,0::gosub 3380:sound
1,7,30:sound 2,516,30
3310 sound 1,516,200:sound 2,897,20
0
3320 fort=1to3:color1,7,5:circle1,1
59,99,47:color1,12,5:circle1,159,99
,47:next
3330 for t=1to4000:next:color 0,2:c
olor
4,7,6:color 1,1:graphic 0,1:end
3340 rem graphic - schlussehrung*
3350 circle 1,159,99,90:circle1,159
,99,89:circle 1,159,99,88
3360 color 1,4,3:circle 1,159,99,63
:circle 1,159,99,62:circle 1,159,99
,61
3370 color 1,8,4:circle 1,159,99,36
:circle 1,159,99,35:circle 1,159,99
,34:return
3380 char 1,20-int((len(n3$)+1)/2),
12,n3$
3390 char 1,2,3,"sieger !"
3400 char 1,30,3,"sieger !"
3410 char 1,2,21,"sieger !"
3420 char 1,30,21,"sieger !"
3430 return
3440 graphic 0,1:color 0,2:color 1,
1:color 4,7,6:end
3450 rem muehle ================p/4
3470 rem 013387 bytes programm  ===
3480 rem 000000 bytes variables ===
3490 rem 000000 bytes arrays    ===
3500 rem 000000 bytes strings   ===
3520 rem ==========================
```

# KARTEN-SPIEL

Das Spiel funktioniert nach den „Bauern-Legen" Regeln. Das Spiel hat 32 Karten. Sie spielen gegen den Computer und erhalten 16 Karten, ebenso der Computer. Wer die meisten Bauern hat, beginnt. Es muß immer mit einem Bauern begonnen werden. Nachdem ein Bauer gelegt wurde, muß in der selben Farbe fortgefahren werden, und zwar in Rück- oder Vorwärtsreihenfolge. Z.B. muß nach einem Herz-Bauer entweder Herz-Dame, König, Ass usw. oder Zehn, Neun, Acht usw. gelegt werden. Hat ein Spieler keine passende Karte mehr, wechselt der Kartenleger. Wenn Sie keine Karte mehr legen können, geben Sie '*' ein.

```
10 rem kartenspiel ==============p4
20 rem (p)        commodore welt
30 rem ============================
40 rem (c)        by cw team      =
50 rem                            =
60 rem                            =
70 rem version 3.5  40z/ascii     =
80 rem plus4 + 1531/1541 (c16/116)=
90 rem ============================

100 wh$=chr$( 5) :re$=chr$( 28)
110 bl$=chr$( 31) :oe$=chr$( 129)
120 lb$=chr$( 153) :db$=chr$( 154)
130 yg$=chr$( 150) :pu$=chr$( 156)
140 ye$=chr$( 158) :s1$=chr$( 32)
150 z1$=chr$( 96) :z3$=chr$( 123)
160 z4$=chr$( 171) :z5$=chr$( 173)
170 z6$=chr$( 174) :z7$=chr$( 176)
180 z8$=chr$( 177) :z9$=chr$( 178)
190 za$=chr$( 179) :zb$=chr$( 189)
200 zc$=chr$( 129) :zd$=chr$( 115)
210 ze$=chr$( 120) :zf$=chr$( 122)
220 zg$=chr$( 98) :zh$=chr$( 168)
230 c1$=chr$( 157) :c2$=chr$( 145)
240 c3$=chr$( 29) :c4$=chr$( 17)
250 cl$=chr$( 147) :he$=chr$( 19)
260 rn$=chr$( 18) :rf$=chr$( 146)
270 forq=1to2:q2$=q2$+c4$:nextq
280 forq=1to4:q4$=q4$+c4$:nextq
290 forq=1to23:qa$=qa$+c4$:nextq
300 forq=1to3:qb$=qb$+z1$:nextq
310 forq=1to6:q6$=q6$+c3$:nextq
320 forq=1to34:qd$=qd$+s1$:nextq
330 forq=1to10:yx$=yx$+zh$:nextq
340 ya$=ya$+z7$+qb$+z9$+qb$+z9$+qb$
+z9$+qb$+z9$+qb$+z9$+qb$+z9$+qb$+z9
$+qb$+z6$
350 yb$=yb$+z4$+qb$+z3$+qb$+z3$+qb$
+z3$+qb$+z3$+qb$+z3$+qb$+z3$+qb$+z3
$+qb$+za$
360 yc$=yc$+z5$+qb$+z8$+qb$+z8$+qb$
+z8$+qb$+z8$+qb$+z8$+qb$+z8$+qb$+z8
$+qb$+zb$
370 yy$=yy$+s1$+c4$+c1$+s1$+c2$
380 yz$=yz$+s1$+s1$+s1$+c4$+c1$+c1$
+c1$+s1$+s1$+s1$+c2$
390 color0,1
400 color4,1
410 printcl$wh$
420 printhe$:fora=1to18:printtab(a)
ye$rn$"Zb"rf$s1$rn$" bauern-legen "
rf$s1$rn$"Sb":nexta
430 printq2$pu$"kennst du die spiel
regel....(j/n)
440 printhe$:fora=1to18:printtab(a)
yg$rn$"Zb"rf$" bauern-legen "rn$"Sb
":nexta
450 geta$:ifa$=""then420
460 ifa$="n"then1790
470 printcl$q4$q6$"ich mische die k
arten"
480 dima$( 33) ,b$( 33) ,c$( 33) ,d$( 4,8)
490 f$=he$+qa$
500 g$=qd$
510 c$( 1)="a":c$( 2)="k":c$( 3)="d":c
$( 4)="b":c$( 5)="z":c$( 6)="9":c$( 7)=
"8
520 c$( 8)="7":b$( 1)=rn$+"A":b$( 2)=r
n$+"S":b$( 3)=rn$+"Z":b$( 4)=rn$+"X"
530 d$=rf$+zg$+c4$+c1$+zg$+c2$:y=16
540 i=1:fora=1to8:forb=1to4:a$( i)=b
$( b)+c4$+c1$+c$( a)+c2$:i=i+1:nextb,
a
550 rem mischen und verteilen ***
560 fora=1to32:b=rnd( 1) :ifb=1thenb=
0.9999
570 c=int( b*( 33-a)) +1:c$( a)=a$( c) :c
1=c:a1$=a$( c)
580 ifc1>32-athen610
590 a$( c1)=a$( c1+1) :c1=c1+1:goto580
600 goto580
610 a$( c1)=a1$:next
620 fora=1to32:b$( a)="":a$( a)="":ne
xta
630 a=0:forb=1to16:a=a+1:a$( b)=c$( a
) :a=a+1:b$( b)=c$( a) :next
640 fora=1to33:c$( a)=yz$:next
650 printcl$oe$:gosub660:goto860
660 printhe$oe$;
670 printtab( 31)rn$"  habe  "
680 printtab( 31)rn$"........":print
tab( 32) c2$rn$s1$y
690 printtab( 31)rn$" karten "re$
700 fora=1to39:printzh$; :nexta:prin
t
710 printtab( 3)bl$ya$
720 b1=0:fora=1to4
730 printtab( 3) ; :forb=1to8:b1=b1+1:
printd$;ye$c$( b1)bl$; :nextb:printd$
:print
740 ifa<4thenprinttab( 3)yb$:nexta
750 printtab( 3)yc$re$
760 fora=1to39:printzh$; :nexta:prin
tlb$c4$"
770 printtab( 3) ; :fora=1to16:printb$
( a)rf$s1$; :nexta:printre$c4$"
780 fora=1to39:printzh$; :nexta:prin
tdb$c4$"
790 printhe$c4$"...................
........"
800 fora=1to16:ifa$( a)>""then830
810 nexta
820 printhe$c4$rn$wh$" ich habe gew
onnen "db$:end
830 fora=1to16:ifasc( b$( a)) <>32then
return
```

```
840 nexta
850 printf$rn$lb$" du hast gewonnen
 "ye$"gratuliere "db$:end
860 fora=1to16:ifmid$(a$(a),5,1)="b
"thenq=q+1
870 nexta
880 q1=int(2*rnd(1)+1)
890 ifq<2then930
900 ifq>2then1280
910 ifq1=1then930
920 ifq1=2then1280
930 printf$"welche karte willst du
legen ";
940 c=0
950 c=c+1
960 geth$(c):ifh$(c)=""then960
970 ifh$(c)="*"thenprintf$g$:goto12
80
980 a=asc(h$(c))
990 ifc=1thengoto1020
1000 rem
1010 rem
1020 fora=1to5:getz$:nexta
1030 ifc<2then950
1040 c=0
1050 fora=1to16
1060 ifasc(mid$(b$(a),2,1))=asc(h$(
1))+128andmid$(b$(a),5,1)=h$(2)then
1100
1070 nexta
1080 printf$"*****  falsche eingabe
 **********":fora=1to999:nexta:pri
ntf$g$
1090 goto930
1100 m$=mid$(b$(a),2,1):n$=mid$(b$(
a),5,1)
1110 ifm$="A"thena1=4:a2=4
1120 ifm$="S"thena1=12:a2=12
1130 ifm$="Z"thena1=20:a2=20
1140 ifm$="X"thena1=28:a2=28
1150 ifn$="a"thena1=a1-2:a2=a2-3
1160 ifn$="k"thena1=a1-1:a2=a2-2
1170 ifn$="d"thena2=a2-1
1180 ifn$="b"then1240
1190 ifn$="z"thena2=a2+1
1200 ifn$="9"thena1=a1+1:a2=a2+2
1210 ifn$="8"thena1=a1+2:a2=a2+3
1220 ifn$="7"thena1=a1+3:a2=a2+4
1230 ifasc(c$(a1))=32then1080
1240 c$(a2)=b$(a)+yy$+b$(a)
1250 printf$g$:b$(a)=yy$
1260 gosub660:goto1280
1270 rem ***** computer *****
1280 fora=1to16:i$=mid$(a$(a),2,1):
j$=mid$(a$(a),5,1)
1290 ifa$(a)=""then1430
1300 ifi$="A"theni=1
1310 ifi$="S"theni=2
1320 ifi$="Z"theni=3
1330 ifi$="X"theni=4
1340 ifj$="a"thenj=1
1350 ifj$="k"thenj=2
1360 ifj$="d"thenj=3
1370 ifj$="b"thenj=4
1380 ifj$="z"thenj=5
1390 ifj$="9"thenj=6
1400 ifj$="8"thenj=7
1410 ifj$="7"thenj=8
1420 d$(i,j)=a$(a)
1430 nexta
1440 z1=9:z=0
1450 z=z+1:ifz=4then1540
1460 fora=1to4:i=a:j=0
1470 ifd$(a,z)>""then1490
1480 nexta:goto1540
1490 forb=4to1step-1:a1=a-1
1500 ifasc(c$(a1*8+b))<>32then1520
1510 ifd$(a,b)>""andasc(c$(a1*8+b+1
))<>32thenj=b:goto1700
1520 nextb
1530 goto1480
1540 z1=z1-1:ifz1=4then1630
1550 fora=1to4:i=a:j=0
1560 ifd$(a,z1)>""then1580
1570 nexta:goto1450
1580 forb=4to8:a1=a-1
1590 ifasc(c$(a1*8+b))<>32then1610
1600 ifd$(a,b)>""andasc(c$(a1*8+b-1
))<>32thenj=b:goto1700
1610 nextb
1620 goto1570
1630 fora=1to4
1640 ifd$(a,4)>""theni=a:j=4:goto17
00
1650 nexta
1660 printhe$c4$"kann keine karte l
egen
1670 fora=1to1500:nexta
1680 printhe$c4$"..................
....."
1690 goto930
1700 y=y-1
1710 printhe$c4$"ich lege  ";
1720 fora=1to16
1730 ifa$(a)=d$(i,j)then1750
1740 nexta
1750 i1=i-1:c$(i1*8+j)=a$(a)+s1$+c4
$+c1$+s1$+c2$+a$(a)
1760 printrn$mid$(a$(a),2,1)mid$(a$
(a),5,1)
1770 a$(a)="":d$(i,j)="
1780 gosub660:goto930
1790 printcl$
1800 printchr$(14)tab(12)wh$"SPIELR
EGELN
1810 printtab(12)yx$
```

```
1820 printc4$"Das  Spiel "ye$"BAUER
N-LEGEN"db$"  hat 32 Karten.
1830 printc4$"davon bekommst  Du  1
6  Karten  und ich
1840 printc4$"16 Karten.
1850 printc4$"Wer  die  meisten  Ba
uern hat, beginnt.
1860 printc4$"Es muss immer mit ein
em Bauern begonnen
1870 printc4$"werden.  Z.B. wenn Du
 PIK-BAUER  legst,
1880 printc4$"dann  kannst  Du  PIK
-DAME, dann   PIK-
1890 printc4$"KOENIG  und danach PI
K-ASS  legen oder
1900 printc4$"von PIK-BAUER, PIK-ZE
HN, NEUN, ACHT bis
1910 printc4$"SIEBEN........."rn$wh
$"Hast  Du  gelesen ?"rf$"  (J)
1920 geta$:ifa$=""then1920
1930 printcl$c4$db$"Gewonnen hat de
r, der zuerst alle seine
1940 printc4$"Karten abgelegt hat.
1950 printc4$"Du kannst  immer nur
 eine Karte legen.
1960 printc4$"Wenn Du  keine Karte
 legen kannst, gib
1970 printc4$"bitte '"wh$"*"db$"' e
in, sonst:
1980 printc4$"herz = "wh$"s"db$" :
pik...= "wh$"a"db$"..."
1990 printc4$"karo = "wh$"z"db$" :
kreuz = "wh$"x"db$"..."
2000 printc4$ye$"Beispiel"db$": pik
-ass = aa : herz-zehn = sz
2010 printc4$"..........karo-ass =
za:kreuz-dame = xd
2020 printc4$rn$wh$"KOENNEN WIR BEG
INNEN?"rf$"...............(J)
2030 geta$:ifa$=""then2030
2040 printchr$(142):goto470
2050 rem kartenspiel ===========p4
2060 rem 060671 bytes memory     ==
2070 rem 006055 bytes programm   ==
2080 rem 000000 bytes variables  ==
2090 rem 000000 bytes arrays     ==
2100 rem 000000 bytes strings    ==
2110 rem 054616 bytes fre (0)    ==
2120 rem ==========================
```



```
10 rem kopfdaten           c16/116/p4
20 rem (p)          cbm-revue team  =
30 rem ==============================
40 rem (c)                          =
50 rem by                           =
60 rem bernd welte                  =
70 rem                              =
80 rem c16 / c116 / plus 4          =
90 rem ==============================

100 wh$=chr$(5):rt$=chr$(13)
110 c4$=chr$(17):rn$=chr$(18)
120 he$=chr$(19):de$=chr$(20)
130 re$=chr$(28):c3$=chr$(29)
140 gr$=chr$(30):bl$=chr$(31)
150 s1$=chr$(32):bk$=chr$(144)
160 c2$=chr$(145):rf$=chr$(146)
170 cl$=chr$(147):pu$=chr$(156)
180 c1$=chr$(157):ye$=chr$(158)
190 cy$=chr$(159):s2$=chr$(160)
200 forq=1to3:q3$=q3$+c4$:nextq
210 forq=1to5:q5$=q5$+c4$:nextq
220 forq=1to17:qb$=qb$+s1$:nextq
230 forq=1to14:qc$=qc$+s1$:nextq
240 forq=1to11:qd$=qd$+s1$:nextq
250 cu=65520:ze=2035:sp=2036
260 printcl$rn$;
270 print"***************************
***************";
280 print"*.........................
..............*";
290 print"*..........a&b welte soft
ware..........*";
300 print"*.........................
..............*";
310 print"***************************
***************";
320 printrf$"...................
................";
330 print"..............***********
..............";
340 print"..............*kopfdaten*
..............";
350 print"..............***********
..............";
360 print"........................
..............";
370 print"........................
..............";
380 print"........................
..............";
390 print"...zum finden der anfangs
adressen von...";
400 print"........................
..............";
410 print"...disketten- und kassett
enprogrammen...";
420 print"........................
```

```
...............";
430 print"...........................
...............";
440 print"...........................
...............";
450 print"............(1) diskette.
...............";
460 print"...........................
...............";
470 print"............(2) kassette.
...............";
480 print"...........................
...............";
490 print"...........................
...............";
500 getkeya$
510 ifa$="1"then530
520 ifa$="2"then760
530 printcl$rn$".............disket
ten-daten............";rf$
540 printq3$qc$rn$"programmname":pr
int:inputpr$
550 open8,8,8,pr$:get#8,a$,b$:close
8
560 a=asc(a$+chr$(0)):b=asc(b$+chr$
(0)):d=a+256*b:gosub650
570 pokeze,10:pokesp,0:syscu
580 print".......startadressen des
programm.......";
590 printc4$c4$qd$"  dezimal="d
600 printc4$qb$"hex= "
610 printq5$"..........neue daten ?
 (j/n)............";
620 getkeyx$
630 ifx$="j"then530
640 ifx$="n"then260
650 pokeze,14:pokesp,22:syscu
660 b=int(d/256):gosub700
670 printh$;
680 b=d-b*256:gosub700
690 printh$;
700 l=band15:h=(band240)/16
710 ifh>9thenh=h+55:goto730
720 h=h+48
730 ifl>9thenl=l+55:goto750
740 l=l+48
750 h$=chr$(h)+chr$(l):return
760 printcl$rn$".............bandko
pf-daten.............";rf$
770 z=z+1
780 open1,1,0
790 t=peek(819)
800 ift=1thent$="basic pgm"
810 ift=3thent$="masch. pgm"
820 ift=4thent$="daten-file"
830 a=peek(820)+peek(821)*256
840 e=peek(822)+peek(823)*256
850 l=e-a
860 n$="":fori=824to1010
870 n$=n$+chr$(peek(i))
880 next
890 printq5$"..........."z".programm
"
900 print:print" typ      :";t$
910 print:ift<>4thenprint" laenge..
:";l"byte":print
920 print" anf.adr.:";a
930 printc4$" name     :";n$
940 printq5$"..........weitere date
n? (j/n)..........";
950 getkeya$
960 ifa$="j"thenclose1:goto760
970 ifa$="n"thenclose1:goto260
980 rem kopfdaten          16/116/p4
990 rem 60671 bytes memory       ==
1000 rem 03557 bytes programm    ==
1010 rem 00000 bytes variables   ==
1020 rem 00000 bytes arrays      ==
1030 rem 00000 bytes strings     ==
1040 rem 57474 bytes free        ==
1050 rem ======================
```

# KOPFDATEN

Dieses Programm sucht die Anfangsadressen von Programmen, welche auf Diskette oder Kassette gespeichert sind.
Nach der Wahl des gewünschten Speichermediums, Kassette oder Diskette, muß der Programmname eingegeben werden. Der Rechner sucht nun das gewünschte Programm und gibt die dezimalen und hexadezimalen Startadressen aus. Nachdem dies geschehen ist, können entweder neue Diskettenprogramme gesucht werden oder Kopfdaten von neuem begonnen werden. Bei Abfrage von Kassettenprogrammen wird zusätzlich eine Information über die Programmart gegeben: Maschinenprogramm, Basicprogramm oder Datenfile.

# VIDEODATEI

Das Spezialprogramm für alle Videofreaks, die endlich einmal Ordnung in ihren Videokassettenhaufen bringen wollen. Auf dem Monitor werden die Menüs dargestellt, unter denen dann der gewünschte Programmpunkt ausgewählt wird. Sie können nach beliebigen Gesichtspunkten wie Titel, Filmart oder Regisseur sortieren und auch einzelne Daten abfragen. Damit Sie nicht jedesmal das Programm zu laden brauchen, wenn Sie bestimmte Informationen über Ihr Videoarchiv benötigen, ist im Programm eine Druckroutine eingebaut, mittels welcher Sie sich Ihr gesamtes Archiv, aber auch lediglich einzelne Datensätze in sauberer Form, auf den Drucker ausgeben lassen können.

```
10 rem videodatei ================
20 rem (p) by    commodore welt   =
30 rem ===========================
40 rem (c) by                     =
50 rem peter ludwig               =
60 rem                            =
70 rem version 3.5  40z/ascii     =
80 rem c16/plus 4+1531/1541+druck.=
90 rem ===========================
100 c4$=chr$(17):re$=chr$(28)
110 gr$=chr$(30):bk$=chr$(144)
120 oe$=chr$(129):br$=chr$(149)
130 lr$=chr$(151):bg$=chr$(152)
140 db$=chr$(152)
150 rn$=chr$(18):rf$=chr$(146)
160 z1$=chr$(108):z2$=chr$(111)
170 z3$=chr$(112):z4$=chr$(186)
180 z5$=chr$(165):z6$=chr$(167)
190 z7$=chr$(42):z8$=chr$(61)
200 z9$=chr$(183):za$=chr$(175)
210 forq=1to5:q5$=q5$+c4$:nextq
220 forq=1to17:qa$=qa$+z9$:nextq
230 forq=1to8:q8$=q8$+c4$:nextq
240 forq=1to17:qb$=qb$+za$:nextq
250 b1$=b1$+z2$+qa$+z3$
260 b2$=b2$+z5$+y8$
270 b3$=b3$+z1$+qb$+z4$
280 color 0,8
290 color 4,16,5
300 dim d$(50,7):remvideodatentabel
le
310 pt$(1)=" datei laden       "
320 pt$(2)=" datei speichern "
330 pt$(3)=" daten eingeben    "
340 pt$(4)=" daten aendern     "
350 pt$(5)=" daten loeschen    "
360 pt$(6)=" daten ausgeben    "
370 pt$(7)=" programm beenden"
380 f$(1)= "videosystem: "
390 f$(2)= "kassette nr. "
400 f$(3)= "titel       : "
410 f$(4)= "zaehlwerk   : "
420 f$(5)= "regie       : "
430 f$(6)= "laufzeit    : "
440 f$(7)= "filmart     : "
450 key 1,"v":rem vorwearts
460 key 2,"r":rem rueckwearts
470 key 3,"a":rem aendern
480 key 8,"m":rem menue
490 goto 810
500 rem *********************
510 rem           programmkopf
520 rem *********************
530 print chr$(147);
540 for i=1 to 40:printrn$db$ ;:nex
t
550 printrn$re$"..........videover
waltung.............";
560 for i=1 to 40:printrn$db$;:next
570 print:print
580 return
590 rem *********************
600 rem          fehlermeldung
610 rem *********************
620 char ,1,24,fe$
630 for i=1 to 1000:next
640 char ,1,24,"
650 return
660 rem *********************
670 rem koepfe programmmteile
680 rem *********************
690 gosub 500
700 char ,10,5,b1$
710 char ,10,6,b2$+pt$(f)+z6$
720 char ,10,7,b3$
730 return
740 rem *********************
750 rem      daten im rechner
760 rem *********************
770 fe$="keine daten im rechner!"
780 gosub 590
790 return
800 next y
810 gosub 500
820 print"=*=*=*=*=*=*=*=*=*=*=*=
*=*=*=*=*=*=*=*"
830 printrn$re$"..........programmf
unktion:............"
840 print"=*=*=*=*=*=*=*=*=*=*=*=
*=*=*=*=*=*=*=*"
850 print
860 printrn$db$c4$"-1- videodaten l
aden ..................."
870 printrn$gr$c4$"-2- videodaten s
peichern................"
880 printrn$re$c4$"-3- videodaten e
ingeben................."
890 printrn$bg$c4$"-4- videodaten a
endern.................."
900 printrn$oe$c4$"-5- videodaten l
oeschen................."
910 printrn$br$c4$"-6- videodaten a
usgeben................."
920 printrn$lr$c4$"-7- programm bee
nden ..................."
930 print
940 printchr$(130)rn$db$rf$;:input
"ihre wahl (1-7): ";f
950 if f=o or f>7 thenprintrn$db$rf
$fe$="diese zahl ist ungueltig !":g
osub 590:goto 810
960 on f goto 980,1180,1390,1610,18
30,2140,2490
970 printrn$db$:input "neuer inhalt
:";d$(zz,x)
980 rem *********************
```

```
990 rem          datei laden
1000 rem*********************
1010 gosub 660:print:print
1020 printrn$re$".....bitte datenka
ssette einlegen und...";
1030 print
1040 printrn$re$"...........zurueck
spulen!..............";
1050 print
1060 printrn$db$:input"danach retur
n druecken..................";x$
1070 open 1,1,0,"videodatei"
1080 input#1,z
1090 for y=1 to z
1100 for i=1 to 7
1110 input#1,d$(y,i)
1120 next i
1130 next y
1140 close 1
1150 print:printrn$db$"daten sind g
eladen!....................."
1160 for x=1 to 2000:next
1170 goto 810
1180 rem *********************
1190 rem         datei speichern
1200 rem *********************
1210 gosub 660:print:print
1220 if z=o then gosub 740:goto 810
1230 printrn$db$"bitte cassette ein
l.und................"
1240 print
1250 printrn$db$"zurueckspulen!....
......................"
1260 print
1270 printrn$db$:input"danach retur
n druecken..................";x$
1280 open 1,1,1,"videodaten"
1290 print#1,z
1300 for y=1 to z
1310 for i=1 to 7
1320 print# 1,d$(y,i)
1330 next i
1340 next y
1350 close 1
1360 print:printrn$db$"daten sind g
esichert!..................."
1370 for x=1 to 2000:next
1380 goto 810
1390 rem ***********************
1400 rem    videodaten eingeben
1410 rem ***********************
1420 gosub 660
1430 z=z+1
1440 print:print
1450 for i=1 to 7
1460 printrn$db$rf$f$(i);
1470 input d$(z,i)
1480 next
1490 print:print
1500 printrn$re$"daten richtig eing
egeben....(j/n).......";
1510 x$="":input x$
1520 if x$="j" then 1550
1530 if x$="n" then z=z-1:goto 1390
1540 print chr$(145);:goto 1500
1550 printrn$db$"noch freier speich
erplatz"rf$:print fre(0)
1560 printrn$re$c4$"weitere eingabe
n(j/n)..................";
1570 x$="":input x$
1580 if x$="j" then 1390
1590 if x$="n" then 810
1600 print chr$(145);:goto 1550
1610 rem ************************
1620 rem         videodaten aendern
1630 rem ************************
1640 zz=1
1650 gosub 660
1660 print:print
1670 if z=0 then gosub 740:goto 810
1680 for i=1 to 7
1690 printi;rn$db$rf$f$(i);d$(zz,i)
1700 next
1710 get x$:if x$="" then 1710
1720 if x$="m" then 810
1730 if x$="v" and zz<z then zz=zz+
1:goto 1650
1740 if x$="r" and zz>1 then zz=zz-
1:goto 1650
1750 if x$="a" then 1770
1760 fe$="............eingabefehle
r!":gosub 590:goto 1710
1770 print:print
1780 printrn$db$:input "feldnummer(
1-7).........................";x
1790 if x<1 or x>7 then print chr$(
145);:goto 1780
1800 print:print
1810 printrn$db$:input "neuer inhal
t:...........................";d$(z
z,x)
1820 goto 1650
1830 rem ***********************
1840 rem    videodaten loeschen
1850 rem ***********************
1860 gosub 660
1870 print:print
1880 if z=0 then gosub 740 :goto 81
0
1890 printrn$db$:input"cassetten nr
.:..........................";d1$
1900 printrn$db$:input"titel:......
...........................";d2$
1910 x=1
1920 if d$(x,2)=d1$ and d$(x,3)=d2$
 then 1960
```

```
1930 if x<z then x=x+1:goto 1920
1940 fe$="titel nicht gefunden!"
1950 gosub 590:goto 810
1960 gosub 660:print:print
1970 for i=1 to 7
1980 print f$(i);d$(x,i)
1990 next
2000 print:print
2010 printrn$re$:input"filmtitel lo
eschen(j/n)...............";x$
2020 if x$="j" then 2050
2030 if x$="n" then 810
2040 print chr$(145);:goto 2010
2050 for y=x to z-1
2060 for i=1 to 7
2070 d$(y,i)=d$(y+1,i)
2080 next i
2090 next y
2100 z=z-1
2110 goto 810
2120 dload"*
2130 dload"*
2140 rem *********************
2150 rem   videodaten ausgeben
2160 rem *********************
2170 gosub 660:print:print
2180 if z=0 then gosub 650:goto 810
2190 printrn$db$:input"drucker oder
 bildschirm(d/b)...........";g$
2200 if g$="d"then open 1,4
2210 gosub 660:print:print
2220 printrn$db$"suchbegriffe:"
2230 print"=*=*=*=*=*=*="
2240 print
2250 for i=1 to 7
2260 s$(i)=""
2270 printrn$db$rf$f$(i);
2280 input s$(i)
2290 next
2300 for y=1 to z
2310 s=0
2320 for i=1 to 7
2330 if s$(i)=""or s$(i)=d$(y,i)the
n s=s+1
2340 next i
2350 if s<>7 then 2430
2360 if g$="d"then print#1
2370 if g$="b"then gosub 660:print
2380 for i=1 to 7
2390 if g$="b" thenprintrn$db$rf$f$
(i);d$(y,i)
2400 if g$="d" then print#1,f$(i);d
$(y,i)
2410 next i
2420 print:printrn$re$q8$:input"dru
ecken return";x$
2430 next y
2440 gosub 660:print:print
2450 printrn$db$q5$"=*=*=*=*=*=*=*d
ateiende=*=*=*=*=*=*=*=*="
2460 print:printrn$re$q8$:input"dru
ecken sie return";x$
2470 if g$="d"then close 1
2480 goto 810
2490 rem **********************
2500 rem       programm beenden
2510 rem **********************
2520 gosub 660:print:print
2530 if z=0 then 2640
2540 printrn$db$"sind alle daten ge
sichert...(j/n).......";
2550 input x$
2560 if x$="n" then 810
2570 if x$="j" then 2590
2580 print chr$(145);:goto 2540
2590 gosub 660:print:print
2600 printrn$db$c4$"das progr.kann
mit goto................";
2610 printrn$db$c4$"500..gestartet
werden.................."
2620 printrn$db$c4$"ohne datenverlu
ste!...................."
2630 print:print
2640 end
2650 rem videodatei ===========c16
2660 rem 012277 bytes memory   ===
2670 rem 007231 bytes programm ===
2680 rem 000000 bytes variables===
2690 rem 000000 bytes arrays   ===
2700 rem 000000 bytes strings  ===
2710 rem 005046 bytes free(0)  ===
2720 rem =========================
```

# KAUFMANN

Dieses Programm stellt eine hervorragende Hilfe für all diejenigen dar, die Auftragsbestätigungen, Lieferscheine, Rechnungen, Mahnungen oder Adreßaufkleber erstellen müssen. Mit Hilfe eines Druckers können Sie nun optisch einwandfreie Dokumente erstellen, um diese für geschäftliche Zwecke zu verwenden. Das Programm verwendet eine Menü-orientierte Eingabemaske, bei der alle erforderlichen Daten abgefragt und zur Ausgabe aufbereitet werden. Zur Errechnung einer Gesamtsumme oder der Mehrwertsteuer beispielsweise brauchen Sie lediglich die Warenmenge und den Einzelpreis einzugeben, die Zwischen-, End- und Mehrwertsteuerwerte werden automatisch errechnet und auf das Formblatt gedruckt. Weiterhin besteht die Möglichkeit, zu jedem Warenpunkt einen Text hinzuzufügen, die Versandform zusätzlich abzudrucken usw. Die einzelnen Menüs sind übersichtlich gehalten und leicht verständlich. Bitte beachten Sie: In die Listingszeilen, in welchen die Bankform (gekennzeichnet mit Punkten) gespeichert ist, müssen Sie Ihre persönlichen Konten und BLZ-Daten einfügen, damit diese auf den Belegen verwendet werden können. »

```
10 rem kaufmann ============c16/p4
20 rem (p)        commodore welt  6
30 rem ============================
40 rem (c)                        =
50 rem a.huebner                  =
60 rem                            =
70 rem version 3.5  40z/ascii     =
80 rem c16/plus4 + 1531/1541 +dru-
90 rem ============================

100 cl$=chr$(47):s1$=chr$(32)
110 z1$=chr$(45):z2$=chr$(61)
120 z3$=chr$(164):z4$=chr$(42)
130 forq=1to21:q1$=q1$+z1$:nextq
140 forq=1to25:q2$=q2$+z1$:nextq
150 forq=1to27:q3$=q3$+z1$:nextq
160 forq=1to38:q4$=q4$+z3$:nextq
170 forq=1to38:q0$=q0$+s1$:nextq
180 forq=1to40:q5$=q5$+z1$:nextq
190 forq=1to47:q6$=q6$+z1$:nextq
200 forq=1to5:q7$=q7$+s1$:nextq
210 qa$=qa$+z4$+q0$+z4$
220 color 0,1 : color 4,1
230 printchr$(159)
240 close 4,4
250 let c=0
260 let sus=0
270 printcl$:fort=1to40:print"*";:n
ext
280 printqa$
290 print"*......programm zur erste
llung von.....*"
300 printqa$
310 print"*......auftragsbestaetigu
ngen..........*"
320 printqa$
330 print"*......rechnungen / liefe
rschein.......*"
340 printqa$
350 print"*......mahnbrief.........
..............*"
360 printqa$
370 print"*......adressaufkleber...
..............*"
380 printqa$
390 for t = 1 to 40 : print "*";:ne
xt
400 printqa$
410 print"*.....-  copy by a. huebn
er  -.........*"
420 printqa$
430 printqa$
440 for t=1 to40: print "*";:next
450 print"*  erstellt auf commodore
 plus/4.......*"
460 for t =1to40: print "*";:next
470 print:for t =1to 40: print "*";
:next
480 for a = 1 to 3500 : next a : pr
int"S" : goto 2650
490 :
500 :
510 rem eingabe
520 :
530 printcl$ : input"tagesdatum (tt
.mm.jjjj)";t$
540 printcl$ : input"anrede des kun
den";a$
550 printcl$ : input"vorname und na
chname";n$
560 printcl$ : input"strasse und ha
usnr.";s$
570 printcl$ : input"plz mit wohnor
t";w$
580 if a =  1  goto 2890 : if a = 3
 goto 2930
590 if ad = 1 goto 1480
600 printcl$ : if wahl$ = "1" then
input"lieferzeitangabe ca. ... ";li
ef$
610 printcl$ : if wahl$ = "1" then
input"auftrags-nr.: ";auf$
620 printcl$ : if wahl$ = "2" then
input"rechnungs.-nr.";r$
630 print"eingabe zur artikelbezeic
hnung":print:print
640 input"anzahl der artikel (max.5
) ";cx$
650 if cx$ = "1" then gosub 710:got
o 780
660 if cx$ = "2" then gosub 710:gos
ub720 : goto 780
670 if cx$ = "3" then gosub 710:gos
ub720:gosub730:goto 780
680 if cx$ = "4" then gosub 710:gos
ub720:gosub730:gosub740:goto780
690 if cx$ = "5" then gosub 710:gos
ub720:gosub730:gosub740:gosub750:go
to780
700 printcl$
710 input"text artikel 1 ";b$ : ret
urn
720 input"text artikel 2 ";b1$: ret
urn
730 input"text artikel 3 ";b2$: ret
urn
740 input"text artikel 4 ";b3$: ret
urn
750 input"text artikel 5 ";b4$: ret
urn
760 :
770 :
780 printcl$
790 print"eingabe zur artikelmenge
"
800 if cx$ = "1" then gosub 850 : g
```

```
oto900
810 if cx$ = "2" then gosub 850:gos
ub860:goto900
820 if cx$ = "3" then gosub 850:gos
ub860:gosub870:goto900
830 if cx$ = "4" then gosub 850:gos
ub860:gosub870:gosub880:goto900
840 if cx$ = "5" then gosub 850:gos
ub860:gosub870:gosub880:gosub890:go
to900
850 input"menge artikel 1";o1:retur
n
860 input"menge artikel 2";l2:retur
n
870 input"menge artikel 3";g3:retur
n
880 input"menge artikel 4";f4:retur
n
890 input"menge artikel 5";y5:retur
n
900 :
910 printcl$
920 print"eingabe zum einzelpreis j
e artikel":print
930 if cx$ = "1" then gosub 980: go
to1060
940 if cx$ = "2" then gosub 980:gos
ub990:goto1060
950 if cx$ = "3" then gosub 980:gos
ub990:gosub1000: goto 1060
960 if cx$ = "4" then gosub 980:gos
ub990:gosub1000:gosub1010: goto 106
0
970 if cx$ = "5" then gosub 980:gos
ub990:gosub1000:gosub1010:gosub1020
: goto1060
980 input"einzelpreis artikel 1";e1
s:return
990 input"einzelpreis artikel 2";e2
s:return
1000 input"einzelpreis artikel 3";e
3s:return
1010 input"einzelpreis artikel 4";e
4s:return
1020 input"einzelpreis artikel 5";e
5s:return
1030 :
1040 :
1050 :
1060 printcl$:printq5$:print
1070 input"kundenversandkosten ";vk
1080 :
1090 input"versandform ";up$
1100 input"zahlungsbedingungen";z$
1110 rem ---------------------
1120 rem berechnung einzelwerte
1130 x1=v+o1
1140 x2=v+l2
1150 x3=v+g3
1160 x4=v+f4
1170 x5=v+y5
1180 a=e1s*x1
1190 b=e2s*x2
1200 c=e3s*x3
1210 d=e4s*x4
1220 e=e5s*x5
1230 :
1240 rem --------
1250 rem endsumme
1260 rem --------
1270 x6=a+b+c+d+e+vk
1280 x7=x6*100/114
1290 x8=x6-x7
1300 printcl$:print:print:print:pri
nt
1310 print " es erfolgt der ausdruc
k !"
1320 printq2$
1330 if wahl$ = "1" then open 4,4
1340 if wahl$ = "2" then open 4,4
1350 :
1360 :
1370 rem chr$( 14).sperrschrift ein
1380 print#4,chr$( 14)"-----------.
............"
1390 print#4,chr$( 14)"
1400 print#4,chr$( 15)"
                         ........."
1410 print#4,chr$( 15)"------------
--------............................
......"
1420 print#4,chr$( 15)"------------
--------------...................."
1430 print#4,chr$( 15)"
1440 print#4,chr$( 15)"
1450 print#4,chr$( 15)"
1460 print#4,chr$( 15)"
1470 print#4,chr$( 15)"
1480 print#4,chr$( 10)
1490 print#4,chr$( 16)"06";a$
1500 print#4,chr$( 16)"06";n$
1510 print#4,chr$( 16)"06";s$
1520 print#4,chr$( 15)"
1530 print#4,chr$( 16)"06";w$
1540 print#4,chr$( 15) : if ad = 1 g
oto 3390
1550 print#4,chr$( 15)"
1560 print#4,chr$( 15)"
1570 print#4,chr$( 15)"
1580 print#4,chr$( 15)"
1590 if ma = 1 then 1640
1600 rem schreiben rechnungsnr.
1610 if wahl$ = "1" then print#4,ch
r$( 14)"auftragsbestaetigung nr.:.."
;auf$
1620 if wahl$ = "1" then print#4,ch
```

```
r$( 15)"lieferzeit ca. .. ";lief$
1630 if wahl$ = "2" then print#4,ch
r$( 14)"rechnung-nr.:";r$
1640 print#4,chr$( 15)"
1650 print#4,chr$( 15)"
1660 rem kopfzeile
1670 print#4,chr$( 10);
1680 print#4,chr$( 16)"64";t$"
1690 print#4,chr$( 15)"
1700 if ma = 1 then 3010
1710 print#4,chr$( 15)"
1720 print#4,chr$( 15)"
1730 print#4,chr$( 10);
1740 print#4,chr$( 16)"00bezeichnung
";chr$( 16)"25menge";chr$( 16)"46einz
el";chr$( 16)"69dm"
1750 print#4,chr$( 16)"46preis";chr$
( 16)"69gesamt"
1760 print#4,chr$( 14)q4$
1770 print#4,chr$( 15)"
1780 print#4,chr$( 15)"
1790 if wahl$ = "2" then print#4,ch
r$( 15)"sie erhalten heute per ";up$
1800 print#4,chr$( 15)"
1810 print#4,chr$( 16)"00---"
1820 print#4,chr$( 15)"
1830 if cx$ = "1" then gosub 1890 :
 goto 2240
1840 if cx$ = "2" then gosub 1890:g
osub1960 : goto 2240
1850 if cx$ = "3" then gosub 1890:g
osub1960 : gosub 2030 : goto 2240
1860 if cx$ = "4" then gosub 1890:g
osub1960 : gosub 2030 : gosub 2100
goto 2240
1870 if cx$ = "5" then gosub 1890:g
osub1960:gosub2030:gosub2100:gosub2
170:goto2240
1880 print#4,chr$( 10)"
1890 print#4,chr$( 16)"00";b$;
1900 print#4,chr$( 16)"25";
1910 print#4,using "##";o1;
1920 print#4,chr$( 16)"46";
1930 print#4,using "#####.##-";e1s;
1940 print#4,chr$( 16)"69";
1950 print#4,using "#####.##-";a :
return
1960 print#4,chr$( 16)"00";b1$;
1970 print#4,chr$( 16)"25";
1980 print#4,using "##";l2;
1990 print#4,chr$( 16)"46";
2000 print#4,using "#####.##-";e2s;
2010 print#4,chr$( 16)"69";
2020 print#4,using "#####.##-";b :
return
2030 print#4,chr$( 16)"00";b2$;
2040 print#4,chr$( 16)"25";
2050 print#4,using "##";g3;
2060 print#4,chr$( 16)"46";
2070 print#4,using "#####.##-";e3s;
2080 print#4,chr$( 16)"69";
2090 print#4,using "#####.##-";c :
return
2100 print#4,chr$( 16)"00";b3$;
2110 print#4,chr$( 16)"25";
2120 print#4,using "##";f4;
2130 print#4,chr$( 16)"46";
2140 print#4,using "#####.##-";e4s;
2150 print#4,chr$( 16)"69";
2160 print#4,using "#####.##-";d :
return
2170 print#4,chr$( 16)"00";b4$;
2180 print#4,chr$( 16)"25";
2190 print#4,using "##";y5;
2200 print#4,chr$( 16)"46";
2210 print#4,using "#####.##-";e5s;
2220 print#4,chr$( 16)"69";
2230 print#4,using "#####.##-";e :
return
2240 print#4,chr$( 16)"30zzgl. versa
ndkosten";
2250 print#4,chr$( 16)"69";
2260 print#4,using "#####.##-";vk
2270 print#4,chr$( 10)"
2280 print#4,chr$( 16)"30rechnungsbe
trag incl  14% mwst";
2290 print#4,chr$( 16)"69";
2300 print#4,using "#####.##-";x6
2310 print#4,chr$( 16)"30"q6$
2320 print#4,chr$( 15)"
2330 print#4,chr$( 10)"
2340 print#4,chr$( 16)"10zahlungsbed
ingung : ";z$
2350 print#4,chr$( 10)"
2360 print#4,chr$( 16)"20 *** mwst.-
satz 14% = dm";
2370 print#4,chr$( 16)"50";
2380 print#4,using "#####.##-";x8
2390 print#4,chr$( 10)"
2400 print#4,chr$( 16)"05wir danken
fuer ihren auftrag ";
2410 print#4,chr$( 16)"32und freuen
uns auf ihren naechsten !"
2420 print#4,chr$( 16)"
2430 print#4,chr$( 16)"
2440 print#4,chr$( 16)"
2450 print#4,chr$( 16)"
2460 print#4,chr$( 16)"
2470 print#4,chr$( 10)"
2480 print#4,chr$( 16)"00volksbank .
..............";
2490 print#4,chr$( 16)"27kreissparka
sse ..........";
2500 print#4,chr$( 16)"55postgiroamt
 ........"
2510 print#4,chr$( 16)"00blz ... ...
```

```
..";
2520 print#4,chr$( 16)"27blz ... ...
..";
2530 print#4,chr$( 16)"55blz ... ...
.."
2540 print#4,chr$( 16)"00kto: ......
.";
2550 print#4,chr$( 16)"27kto: ......
.";
2560 print#4,chr$( 16)"55kto: ......
-..." : close 4,4 : print"S"
2570 if wahl$ = "1" then input"copi
e der auftragsbestaetigung ";c$
2580 if wahl$ = "2" then input"copi
e der rechnung (j/n)";c$
2590 if  c$="j"then 1330 (:else2600
)
2600 if wahl$ = "2" then input"weit
ere rechnungen (j/n)";i$
2610 if i$ = "j" goto 540 (:else
2630)
2620 if wahl$ = "1" then input"weit
ere auftragsbestaetigungen ";i$
2630 close 4,4 : return
2640 if i$="n"then return
2650 print"S":rem menueauswahl ****
2660 for t = 1 to 40 : print "*";:n
ext
2670 print"*.....m e n u e  -  a u
s w a h l......*"
2680 printqa$
2690 printz4$q7$q3$q7$s1$z4$
2700 printqa$
2710 printqa$
2720 print"*..auftragsbestaetigunge
n.....=...1....*"
2730 printqa$
2740 print"*  rechnung/lieferschein
......=...2....*"
2750 printqa$
2760 print"*..mahnbrief...........
......=...3....*"
2770 printqa$
2780 print"*..adressaufkleber......
......=...4....*"
2790 printqa$
2800 print"*..ende des programms...
......=...5....*"
2810 for t = 1 to 40 : print"*";: n
ext
2820 print:print:print:print:input"
...bitte waehlen sie...........";w
ahl$
2830 let a = 0 : let ma = 0 : let a
d = 0
2840 if wahl$ = "1" then gosub 530:
goto2650
2850 if wahl$ = "2" then gosub 530:
goto2650
2860 if wahl$ = "3" then a = a + 1
: gosub 530 : goto 2650
2870 if wahl$ = "4" then print"S" :
 goto 3300
2880 if wahl$ = "5" then print"S":g
oto 3420
2890 printcl$:for t = 1 to 40 : pri
nt"*";:next:print
2900 print"*........mahnbrief......
.................*"
2910 for t = 1 to 40 : print"*";:ne
xt:print:print:print"bitte eingeben
:"
2920 input"mahnstufe...........";m
ahns$
2930 input"rechnungs-nr.:.......";r
$
2940 input"erstdatum 'rechnung' ";a
r$
2950 input"mahnbetrag (dm)......";b
$
2960 printcl$:print:print:print:pri
nt"es erfolgt der ausdruck des mahn
briefes."
2970 printq5$
2980 open 4,4 : let ma = 0
2990 ma = ma + 1 : goto1380
3000 :
3010 rem*********************
3020 remtext fuer mahnbrief
3030 rem*********************
3040 print#4,chr$( 16)"01buchhaltung
, konto-nr.: ";r$
3050 print#4,chr$( 16)"01mahnstufe =
  ";mahns$
3060 print#4:print#4:print#4,chr$( 1
6)"01sehr geehrter herr ";n$;",":pr
int#4
3070 print#4,chr$( 16)"01leider muss
ten wir feststellen, dass der unten
 genannte betrag"
3080 print#4,chr$( 16)"01noch nicht
bei uns eingetroffen ist. sicher ha
ben sie vergessen"
3090 print#4,chr$( 16)"01die ueberwe
isung vorzunehmen.":print#4:print#4
3100 print#4,chr$( 16)"01fuer eine b
aldige erledigung danken wir ihnen
schon im voraus."
3110 print#4:print#4:print#4
3120 print#4,chr$( 16)"01mit freundl
ichen gruessen"
3130 print#4,chr$( 16)"01..........
...":print#4:print#4:print#4
3140 print#4,chr$( 16)"01.........."
:print#4:print#4:print#4:print#4
3150 print#4,chr$( 14):print#4,chr$(
```

```
16)"01a u f s t e l l u n g "
3160 print#4,chr$( 16)"01"q1$:print#
4
3170 print#4,chr$( 15);chr$( 16)"01re
chnung vom ";ar$;chr$( 16)"30rechnun
gs-nr: ";r$;
3180 print#4,chr$( 16)"57betrag in d
m ";
3190 print#4,using "####.##-";b$
3200 for t = 1 to 76 : print#4,chr$
( 16)"01-";:next
3210 print#4:print#4:print#4:print#
4,chr$( 16)"01anlage":print#4,chr$( 1
6)"01------"
3220 print#4:print#4,chr$( 16)"01rec
hnungskopie"
3230 print#4:print#4:print#4
3240 print#4,chr$( 16)"01falls der a
usgleich inzwischen erfolgt ist, be
trachten sie";
3250 print#4,chr$( 16)"60dieses schr
eiben"
3260 print#4,chr$( 16)"01al's erledig
t !":close 4,4
3270 printcl$:input"copie des ausdr
ucks ( j/n) ";ausdru$
3280 if ausdru$ = "j" goto 2980 : g
oto 3270
3290 if ausdru$ = "n" goto 2650
3300 for t = 1 to 40 : print"*";:ne
xt
3310 rem ************************
3320 print"*......adressaufkleber..
.............*"
3330 rem ************************
3340 for t = 1 to 40 : print"*";:ne
xt
3350 open4,4 : let ad=0
3360 print:print:print:print:print"
ihre eingabe bitte !":for t = 1 to
20:print"-";:next
3370 for j = 1 to 500 : next j
3380 print:print:print:ad = ad + 1
: goto 540
3390 printcl$ : ad = 0 : input"weit
ere aufkleber ( j/n) ";aufkl$
3400 if aufkl$ = "j" then ad = ad +
 1:goto 540
3410 if aufkl$ = "n" then close 4,4
 : goto 2650
3420 for t = 1 to 40 : print"*";:ne
xt
3430 print"*......ende des programm
s.............*"
3440 for t = 1 to 40 : print"*";:ne
xt
3450 print:print:print:print"tagesd
atum = ";t$
3460 print:print:print:print"pgm.-l
aenge = ";60671 - fre(x); "in byte"
3470 print:printchr$( 158):printchr$
( 18)"...copy by. a. huebner..08/198
5":printchr$( 159)
3480 rem printchr$( 146)
3490 for t = 1 to 40 : print"*";:ne
xt:print:print
3500 end
3510 rem kaufmann ==============p4
3520 rem 000000 bytes memory      ==
3530 rem 000000 bytes programm    ==
3540 rem 000000 bytes variables   ==
3550 rem 000000 bytes arrays      ==
3560 rem 000000 bytes strings     ==
3570 rem 000000 bytes fre (0)     ==
3580 rem ========================
11100 rem 060671 bytes memory
===
11200 rem 010369 bytes programm
===
11600 rem 050267 bytes fre (0)
===
```

# KÖNIG ARTHUS

Wir warnen dringend vor dem Abtippen dieses Programmes! Sollten Sie trotzdem den dringlichen Wunsch verspüren, einmal König zu werden, behaupten Sie nicht, wir hätten Sie nicht gewarnt. Denn auch bei uns wurde von diesem Superspiel für Tage die Redaktion lahmgelegt und nur unter Mordandrohungen des Verlegers waren wir wieder zur Arbeit zu bewegen.
In diesem Spiel kämpfen Sie sich durch viele Screens hindurch, die von schönen Farbgrafiken untermalt sind.
Die Aufgabe selbst ist schnell erklärt: Durch geschicktes Regieren müssen Sie vor Ihren Mitspielern (bis zu acht sind möglich) König werden. Doch dies gestaltet sich als schwere Übung, denn unzählige Faktoren müssen berücksichtigt werden, um dem Ziel näherzukommen. Steuern müssen festgesetzt, Getreiderationen verteilt, Armeen ausgerüstet und Märkte errichtet werden. Neben diesen Regierungspflichten müssen Sie außerdem auch noch einen regen Handel mit Getreide und Ländereien betreiben, um auch finanziell dem Anspruch eines Königs gerecht zu werden. Doch übernehmen dürfen Sie sich hierbei auch nicht, denn sehr leicht überfallen Sie Ihre grimmigen Mitspieler (oder der Computer!) und alle Bemühungen waren umsonst.
Auch sollten Sie sich nicht zu viel Zeit lassen, denn zur damaligen Zeit währte das Leben eines Monarchen nicht lange, unbekannte Krankheiten in Form von etwas zwielichtigen Damen und dergleichen verursachen leicht den Ehrentod eines Thronanwärters!
Doch mehr sei nicht verraten, denn der Sinn dieses Strategiespieles ist es ja, seine eigene persönliche Taktik zum Erreichen des Zieles aufzustellen.
Wir wünschen viel Erfolg!

```
10 rem koenig arthus         =========p4
20 rem (c) commodore welt team        =
30 rem       t. seibt                 =
40 rem                                =
50 rem ===============================
60 rem version 3.5   40z/ascii        =
70 rem mit cass. oder floppy          =
80 rem plus/4 oder c 16+64kb          =
90 rem ===============================
100 wh$=chr$(5):cv$=chr$(8)
110 ce$=chr$(9):rt$=chr$(13)
120 so$=chr$(14):cd$=chr$(17)
130 rn$=chr$(18):ho$=chr$(19)
140 de$=chr$(20):ee$=chr$(27)
150 re$=chr$(28):ch$=chr$(29)
160 gr$=chr$(30):bl$=chr$(31)
170 sp$=chr$(32):oe$=chr$(129)
180 f0$=chr$(130):f1$=chr$(131)
190 f1$=chr$(133):f3$=chr$(134)
200 f5$=chr$(135):f7$=chr$(136)
210 f2$=chr$(137):f4$=chr$(138)
220 f6$=chr$(139):hp$=chr$(140)
230 sr$=chr$(141):gb$=chr$(142)
240 bk$=chr$(144):cu$=chr$(145)
250 rf$=chr$(146):cl$=chr$(147)
260 in$=chr$(148):br$=chr$(149)
270 hr$=chr$(150):gu$=chr$(151)
280 gz$=chr$(152):hg$=chr$(153)
290 hb$=chr$(154):gd$=chr$(155)
300 pu$=chr$(156):ct$=chr$(157)
310 ye$=chr$(158):cy$=chr$(159)
320 sq$=chr$(160):sr$=chr$(224)
330 z1$=chr$(96):z2$=chr$(98)
340 z3$=chr$(123):z4$=chr$(171)
350 z5$=chr$(173):z6$=chr$(174)
360 z7$=chr$(176):z8$=chr$(177)
370 y1$=chr$(178):y2$=chr$(179)
380 y3$=chr$(189)
390 forq=1to37:q3$=q3$+z1$:nextq
400 forq=1to6:q6$=q6$+z1$:nextq
410 forq=1to7:q7$=q7$+z1$:nextq
420 forq=1to8:q8$=q8$+z1$:nextq
430 forq=1to9:q9$=q9$+z1$:nextq
440 v1$=chr$(58):v2$=chr$(105)
450 v3$=chr$(117):v4$=chr$(127)
460 v5$=chr$(161):v6$=chr$(162)
470 v7$=chr$(168):v8$=chr$(169)
480 v9$=chr$(188)
490 w1$=chr$(190):w2$=chr$(191)
500 goto620
510 forzz=1to40:printbl$chr$(163);:
nextzz:return
520 print" "bk$rn$".........drueck
en sie eine taste...."
530 geta$:ifa$=""then530
540 return
550 printcl$cv$gb$
560 form=1to40:printbl$chr$(175);:n
extm
570 forj=1to40:printye$chr$(164);:n
extj
580 printye$rn$"* commodore welt pr
aesentiert...:......*"rf$
590 printye$rn$"*.............koeni
g arthus............*"rf$
600 print:forn=1to40:printbl$chr$(1
83);:nextn
610 return
620 y(0)=1200
630 databauer,arbeiter,herr,baron,g
raf,herzog,prinz,koenig
640 dataholstein,sachsen,boehmen,sc
hwaben,bayern,schlesien,arelat,pomm
ern
650 restore:color 0,1:color 4,1:gos
ub550
660 fora=1to8:reada$:next
670 printcd$wh$"wieviel spielen mit
 ? <1-8>"
680 print:inputf
690 iff<1orf>8then650
700 print:fora=1tof:readt$(a)
710 printbl$"wer soll in ";t$(a);"
regieren "
720 print:inputn$:print
730 iflen(n$)>10then710
740 n$(a)=n$+" von "+t$(a)
750 g(a)=10
760 h(a)=10
770 i(a)=10
780 j(a)=2
790 o(a)=1220+int(rnd(1)*30)
800 k(a)=1500
810 l(a)=12000
820 r(a)=4000
830 t(a)=1
840 u(a)=1
850 n(a)=5
860 p(a)=25
870 q(a)=5
880 m(a)=20
890 s(a)=1800
900 next
910 fora=1tof
920 restore
930 b=t(a)
940 forc=1tob
950 readt$(a)
960 next
970 next
980 gosub550
990 printcd$ye$"kennen sie die rege
ln bereits (j/n)"
1000 print:inputa$
1010 ifleft$(a$,1)="n"thengosub5720
1020 color 0,1:color 4,1
```

```
1030 gosub550:printspc(45)bl$"1. "w
h$"blutiger anfaenger"cd$
1040 printtab(5)bl$"2. "ye$"gespons
erter amateur"cd$
1050 printtab(5)bl$"3. "pu$"bezahlt
er profi"cd$
1060 printtab(5)bl$"4. "gr$"meister
 aller klassen"
1070 printcd$ye$"ein wie guter spie
ler sind sie <1-4>"
1080 print:inputu
1090 u(0)=int(u)
1100 ifu<1thenu(0)=1
1110 ifu>4thenu(0)=4
1120 u(0)=u(0)+5
1130 e=e+1
1140 ift(e)=-1thene=e+1
1150 ift(1)<1andt(2)<1andt(3)<1andt
(4)<1andt(5)<1andt(6)<1andt(7)<1and
t(8)<1then10
1160 ife>fthene=0:y(0)=y(0)+1:goto1
130
1170 ify(0)>o(e)then1130
1180 ify(0)=o(e)then1360
1190 gosub1630
1200 gosub2360
1210 gosub3140
1220 gosub2770
1230 gosub3800
1240 gosub4660
1250 gosub5040
1260 goto1130
1270 color 0,6,3:color 4,6,3
1280 printcl$bk$rn$" land  hoef- le
ib-  kirche haend- sol-  ";
1290 print" qm/2  linge eigene kler
us ler    daten "cd$
1300 fora=1tof:printgz$t$(a)" "n$(a
)
1310 l$=str$(l(a))
1320 l$=wh$+right$(l$,len(l$)-1)
1330 printl$tab(6)n(a)tab(12)s(a)ta
b(19)q(a)tab(26)m(a)tab(33)p(a)
1340 next
1350 print:gosub520:return
1360 color 0,3,3:color 4,3,3
1370 printcl$rn$bk$v3$;:forp=1to38:
printz1$;:nextp:printv2$;
1380 printz2$" eine schreckliche na
chricht traf ein "z2$;
1390 printchr$(106);:forp=1to38:pri
ntz1$;:nextp:printchr$(107)
1400 printcd$ye$t$(e)" "n$(e)
1410 print"ist gerade "
1420 t(e)=-1:y=int(rnd(8)*8)+1
1430 ify(0)>1250thenprint"nach lang
er regierung gestorben worden.":got
o1490
1440 ify<4thenprint"im schweren sch
neesturm verschollen."
1450 ify=5thenprint"an cholera (vib
rio cholerae) gestorben."
1460 ify=4thenprint"beim sturz vom
pferd leider toedlich....verungluec
kt."
1470 ify=6thenprint"von attentaeter
n ermordet worden."
1480 ify>6thenprint"an unbekannter
krankheit (maetresse)....verstorben
."
1490 print:eg=1
1500 gosub520
1510 gosub3800
1520 gosub1270
1530 eg=0
1540 goto1130
1550 i1=int(i1):return
1560 c1=int(c1):return
1570 s1=int(s1):return
1580 k(e)=int(k(e)):return
1590 z=(int(rnd(a)*a)+1)*s(e)/100:z
2=z:printbl$int(z2)"neugeborene lei
beigene."
1600 print:s(e)=s(e)+z2:return
1610 z=(int(rnd(a)*a)+1)*s(e)/100:z
2=z:printre$int(z2)"leibeigene gest
orben."
1620 print:s(e)=s(e)-z2:return
1630 w=(int(rnd(5)*5)+2+int(rnd(6)*
6))/2
1640 onwgoto1650,1660,1670,1680,169
0
1650 w$="duerre":goto1700
1660 w$="schlechtes wetter":goto170
0
1670 w$="durchschnittliches wetter"
:goto1700
1680 w$="gutes wetter":goto1700
1690 w$="phantastisches wetter":got
o1700
1700 r=int(rnd(50)*50)+1
1710 r(e)=(r(e)*100-r(e)*r)/100
1720 x=l(e):y=(s(e)-d(e)*100)*5
1730 ify<0theny=0:goto1750
1740 ify<xthenx=y
1750 y=r(e)*2:ify<xthenx=y
1760 y=w-.5:h1=x*y:r(e)=r(e)+h1
1770 d1=n(e)*100+c(e)*40+m(e)*30+p(
e)*10+s(e)*5
1780 l=(3*w+int(rnd(6)*6)+2+int(rnd
(6)*6)+10)/10
1790 ifh1<1theny=2:goto1810
1800 y=d1/h1:ify>2theny=2
1810 ify<.8theny=.8
1820 l=l*y:l=int(l*10):l=l/10
1830 z=6-w
```

```
1840 g=(z*5+int(rnd(5)*5)+int(rnd(5
)*5))/5*y*20
1850 return
1860 print
1870 printbr$"von ihrem lagerweizen
 wurde von ratten"
1880 printr" prozent gefressen"
1890 printbl$w$
1900 printbk$"* jahresernte weizen:
"int(h1)"scheffel *"
1910 print:ifk(e)<32766thengosub158
0
1920 r$=str$(int(r(e))):r$=right$(r
$,len(r$)-1):ifr(e)<0thenr$="  0"
1930 d$=str$(int(d1)):d$=right$(d$,
len(d$)-1)
1940 g$=str$(int(g)):g$=" "+g$:l$=s
tr$(l):l$="  "+l$
1950 k$=str$(int(k(e)))
1960 printbk$z7$q7$y1$q6$y1$q7$y1$q
8$y1$q6$z6$;
1970 printz2$"weizen "z2$"weizen"z2
$"weizen "z2$"land...."z2$"$$$$$$"z
2$;
1980 printz2$"reserve"z2$"bedarf"z2
$"preis.."z2$"preis..."z2$"......"z
2$;
1990 printbk$z4$q7$z3$q6$z3$q7$z3$q
8$z3$q6$y2$;
2000 printz2$r$tab(8)z2$d$tab(15)z2
$g$tab(23)z2$l$tab(32)z2$k$tab(39)z
2$;
2010 printbk$z4$q7$z8$q6$z3$q7$z3$q
8$z3$q6$y2$;
2020 printz2$"scheffel......"z2$"10
00 sl"z2$"morgen.."z2$"gold.."z2$;
2030 printbk$z5$q7$z1$q6$z8$q7$z8$q
8$z8$q6$y3$;
2040 return
2050 j=(j(e)*300-500)*t(e)
2060 onj(e)goto2070,2090,2110,2130
2070 j$="gnaedig"
2080 goto2140
2090 j$="fair"
2100 goto2140
2110 j$="ungerecht"
2120 goto2140
2130 j$="bestechl."
2140 y=150-g(e)-h(e)-i(e)
2150 ify<1theny=1
2160 c1=(n(e)*180+q(e)*75+m(e)*20)*
(y/100)+u(e)*100
2170 s1=(n(e)*50+m(e)*25+u(e)*19)*(
y/100)*(5-j(e))/2
2180 i1=n(e)*250+u(e)*20+(10*j(e)*n
(e))*(y/100)
2190 c1=c1*g(e)/100
2200 ifc1<32760thengosub1560
2210 s1=s1*h(e)/100
2220 ifs1<32760thengosub1570
2230 i1=i1*i(e)/100
2240 ifi1<32760thengosub1550
2250 printbk$z7$q3$z6$
2260 printz2$"steuer einkuenfte"j+c
1+s1+i1"goldstuecke."tab(38)z2$
2270 printbk$z4$q9$y1$q7$y1$q9$y1$q
9$y2$
2280 printz2$"zoll....."z2$"verkauf
"z2$"einkommen"z2$"gerichts-"z2$
2290 printz2$"abgaben.."z2$"steuer
"z2$"steuer..."z2$"wesen...."z2$
2300 printbk$z4$q9$z3$q7$z3$q9$z3$q
9$y2$
2310 printz2$".."g(e)ct$"%"tab(10)z
2$" "h(e)ct$"%";
2320 printtab(18)z2$".."i(e)ct$"%"t
ab(28)z2$j$tab(38)z2$
2330 printz2$".."c1tab(10)z2$" "s1t
ab(18)z2$" "i1tab(28)z2$".."jtab(38
)z2$
2340 printbk$z5$q9$z8$q7$z8$q9$z8$q
9$y3$
2350 return
2360 color 0,8,7:color 4,8,7
2370 printcl$bk$t$(e)" "n$(e)
2380 gosub510
2390 gosub1860
2400 printcd$"(1) kaufe weizen....(
3) kaufe land"
2410 print"(2) verkaufe weizen (4)
verkaufe land"
2420 printcd$bl$"(0) weiterspielen.
..(0-4) bitte waehlen"cd$
2430 inputi
2440 ifi>4then2360
2450 ifi<1thenreturn
2460 onigoto2470,2570,2640,2700
2470 printbk$"wieviel weizen wollen
 sie kaufen"cd$
2480 inputi
2490 ifi<1then2520
2500 k(e)=k(e)-(i*g/1000)
2510 r(e)=r(e)+i
2520 printcl$bk$t$(e)" "n$(e)
2530 gosub510
2540 printtab(80)
2550 print:gosub1910
2560 goto2400
2570 printbk$"wieviel weizen wollen
 sie verkaufen"
2580 inputi
2590 ifi<1then2520
2600 ifi>r(e)thenprintre$"angeber,
sie haben ja gar nicht so viel!":go
to2570
2610 k(e)=k(e)+(i*g/1000)
```

```
2620 r(e)=r(e)-i
2630 goto2520
2640 printpu$"wieviele morgen wolle
n sie kaufen"cd$
2650 inputi
2660 ifi<0then2520
2670 l(e)=l(e)+i
2680 k(e)=k(e)-(i*l)
2690 goto2520
2700 printpu$"wieviele morgen wolle
n sie verkaufen"
2710 inputi
2720 ifi<0then2520
2730 ifi>(l(e)-5000)thenprintre$"na
rr ! sie haben ja gar nicht so viel
 !":goto2700
2740 l(e)=l(e)-i
2750 k(e)=k(e)+(i*l)
2760 goto2520
2770 printcl$:color 0,7,3:color 4,7
,3
2780 printye$t$(e)" "n$(e)
2790 gosub510
2800 print:gosub2050
2810 printcd$"(1) zollabgaben.....(
3) einkommensteuer"
2820 print"(2) verkaufsteuer...(4)
gericht"
2830 print"(0) = weiter........"wh$
"(0-4) = waehlen sie"cd$
2840 inputi:print
2850 ifi>4then2830
2860 ifi<1then3100
2870 onigoto2880,2930,2970,3010
2880 printwh$"neue zollabgaben
(0-100)"cd$
2890 inputi
2900 ifi>100theni=100
2910 ifi<0theni=0
2920 g(e)=i:goto2770
2930 printwh$"neue verkaufsteuer (0
-50)"cd$
2940 inputi
2950 if(i>50)or(i<0)theni=5
2960 h(e)=i:goto2770
2970 printwh$"neue einkommensteuer
(0-25)"cd$
2980 inputi
2990 if(i<0)or(i>25)theni=0
3000 i(e)=i:goto2770
3010 printwh$cl$:forq=1to40:printch
r$(175);:nextq
3020 printrn$spc(16)"gericht"
3030
forq=1to40:printchr$(183);:nextq
3040 printcd$"(1) gnaedig.........(
3) ungerecht"
3050 printcd$"(2) fair............(
4) bestechlich"
3060 printcd$bk$"welche stufe"cd$
3070 inputi
3080 if(i>4)or(i<1)theni=1
3090 j(e)=i:goto2770
3100 k(e)=k(e)+c1+s1+i1+j
3110 ifk(e)<0thenk(e)=k(e)*1.5
3120 ifk(e)<(-10000*t(e))then5390
3130 return
3140 printcl$bk$t$(e)" "n$(e)
3150 gosub510
3160 printtab(120):gosub1910
3170 printcd$bl$"wieviel weizen geb
en sie ihren leuten"
3180 print"zum taeglichen verbrauch
"cd$
3190 inputg
3200 ifg<(r(e)/4)thenprintre$"geizk
nochen! mindestens 25% freigeben!":
goto3170
3210 ifg>(r(e)*.75)thenprintre$"vol
lidiot! mindestens 25% aufbewahren!
":goto3170
3220 r(e)=r(e)-g
3230 printcl$bk$t$(e)" "n$(e)
3240 forzz=1to40:printchr$(163);:ne
xtzz
3250 color 0,2,0:color 4,2,0
3260 z=g/d1-1
3270 ifz>0thenz=z/2
3280 ifz>.25thenz=z/10+.25
3290 z2=50-g(e)-h(e)-i(e)
3300 ifz2<0thenz2=z2*j(e)
3310 z2=z2/10
3320 ifz2>0thenz2=z2+3-j(e)
3330 z=z+(z2/10)
3340 ifz>.5thenz=.5
3350 ifg<(d1-1)then3610
3360 a=7:gosub1590
3370 a=3:gosub1610
3380 if(g(e)+h(e))<35thenm(e)=n(e)+
int(rnd(2)*2):q(e)=q(e)+int(rnd(3)*
3)
3390 ifi(e)<int(rnd(20)*20)+1thenn(
e)=n(e)+int(rnd(2)*2):q(e)=q(e)+int
(rnd(3)*3)
3400 ifg<(d1+d1*.3)then3540
3410 z2=s(e)/1000
3420 z=(g-d1)/d1*10
3430 z=z*z2*int(rnd(25)*25+1)+int(r
nd(40)*40+1)
3440 ifz>32000thenz=32000
3450 z2=z:z=int(rnd(4)*z2)+1
3460 printcd$bk$z"leibeigene sind e
ingewandert"cd$
3470 s(e)=s(e)+z
3480 z2=z/5
3490 z=int(rnd(5)*z2)+1
```

```
3500 ifz>50thenz=50
3510 m(e)=m(e)+z
3520 n(e)=n(e)+1
3530 q(e)=q(e)+2
3540 ifj(e)<3then3600
3550 j1=s(e)/100*(j(e)-2)*(j(e)-2)
3560 j1=int(rnd(8)*j1)+1
3570 s(e)=s(e)-j1
3580 printcd$cy$j1"leibeigene sind
wegen ungerechter"
3590 print"herrschaft ausgewandert,
 alter miesling"
3600 goto3670
3610 x=(d1-g)/d1*100-9
3620 ifx>65thenx=65
3630 ifx<0thenx2=0:x=0
3640 a=3:gosub1590
3650 a=x2+8:gosub1610
3660 goto3540
3670 z=a(e)*75
3680 k(e)=k(e)+z
3690 ifz>0thenprintcd$bl$" markt br
achte"z"golddstuecke ein."cd$
3700 ifs(e)<32766thens1=s(e):gosub1
570:s(e)=s1
3710 z=d(e)*(55+int(rnd(250)*250+1)
)
3720 ifz>0thenk(e)=k(e)+z:printpu$"
 muehle brachte"z"goldstuecke ein."
cd$
3730 z=p(e)*3:printgr$" armee koste
t"z"goldstuecke."cd$:k(e)=k(e)-z
3740 if(l(e)/1000)>p(e)then5500
3750 if(l(e)/500)<p(e)then3790
3760 fora=1tof:ifa=ethen3780
3770 ifp(a)>(p(e)*2.4)then5500
3780 next
3790 print:gosub520:return
3800 color 0,6,0:color 4,5,0
3810 printcl$bk$
3820 l2=(l(e)/1000)
3830 ifl2<10thenx=30:y=20:goto3900
3840 ifl2<30thenx=25:y=17:goto3900
3850 ifl2<50thenx=20:y=14:goto3900
3860 ifl2<70thenx=15:y=11:goto3900
3870 ifl2<90thenx=10:y=08:goto3900
3880 ifl2<110thenx=5:y=5:goto3900
3890 x=1:y=2:goto3900
3900 a$=bk$:fora=1toy:a$=a$+cd$:nex
ta
3910 fora=1tox:a$=a$+ch$:nexta
3920 fora=xto39:a$=a$+chr$(118):nex
t
3930 printho$a$
3940 a$=bk$:fora=1toy-1:a$=a$+cd$:n
ext
3950 fora=1tox:a$=a$+ch$:next
3960 b$=a$+ct$:c$=a$:d$=a$:e$=a$:f$
=a$:g$=a$
3970 fora=y-1to23:a$=a$+cd$+ct$+chr
$(118):next
3980 printho$a$;
3990 c1$=br$+cu$+w2$+w2$+w1$+cd$+ct
$+ct$+ct$+rn$+v5$
4000 c2$=sp$+cd$+ct$+ct$+v5$+sp$+rf
$
4010 c1$=c1$+c2$
4020 c3$=br$+cu$+w2$+w2$+w1$+v9$+rn
$+w2$
4030 c4$=w2$+cd$+ct$+ct$+ct$+ct$+ct
$+ct$+v5$+sp$+rf$
4040 c5$=v7$+v7$+rn$+sp$+rf$+v5$+cd
$+ct$+ct$+ct$+ct$+ct$+ct$
4050 c6$=rn$+w1$+p$+sp$+sp$+sp$+sp$
+v9$+rf$
4060 c3$=c3$+c4$+c5$+c6$
4070 if(p(e)-5)<(l(e)/1000)then4110
4080 if(p(e)/2)<(l(e)/1000)then4100
4090 printho$b$c3$:goto4110
4100 printho$b$c1$
4110 fora=y-2toy+int((23-y)*.8)
4120 c$=c$+cd$:next
4130 fora=x+1tox+int((39-x)*.1)
4140 c$=c$+ch$:next:c$=c$+ch$
4150 z=c(e)+1:ifz>7thenz=7
4160 ifz=0then4240
4170 c$(2)=bl$+v9$+rn$+v6$+rf$+w1$+
cu$+ct$+ct$+ct$+rn$+pu$+v8$+sp$+v4$
4180 c$(3)=c$(2)+cu$+ct$+ct$+rf$+wh
$+z3$:c$(1)=" "
4190 c$(4)=c$(3)+cd$+cd$+bl$+rn$+v6
$+v6$+v6$+v6$+rf$+w1$
4200 c$(5)=c$(4)+cu$+ct$+ct$+ct$+rn
$+pu$+v8$+sp$+v4$
4210 c$(6)=c$(5)+cu$+ct$+ct$+rf$+wh
$+z3$
4220 c$(7)=c$(6)+ct$+ct$+ct$+bk$+rf
$+z3$+cd$+ct$+rn$+ye$
4230 c1$=c$(z):printho$;c$;c1$;
4240 fora=y-2toy+int((23-y)*.8)
4250 d$=d$+cd$:next
4260 fora=xtox+int((39-x)*.5)
4270 d$=d$+ch$:next
4280 ifb(e)=0then4350
4290 ifl(e)<10000thend$=d$+ct$+ct$
4300 c$(1)=rn$+bk$+v1$+v2$+cu$+ct$+
ct$+pu$+v8$+v4$
4310 c$(2)=c$(1)+rf$+ye$+v3$+v2$+cd
$+ct$+ct$+br$+rn$+z7$+z6$
4320 c$(3)=c$(2)+br$+rn$+v1$+v2$+cu
$+ct$+ct$+pu$+v8$+v4$
4330 a=b(e):ifa>3thena=3
4340 printho$d$c$(a)
4350 fora=ytoy+int((25-y)*.3)
4360 e$=e$+cd$:next
4370 fora=xtox+int((39-x)*.1)
```

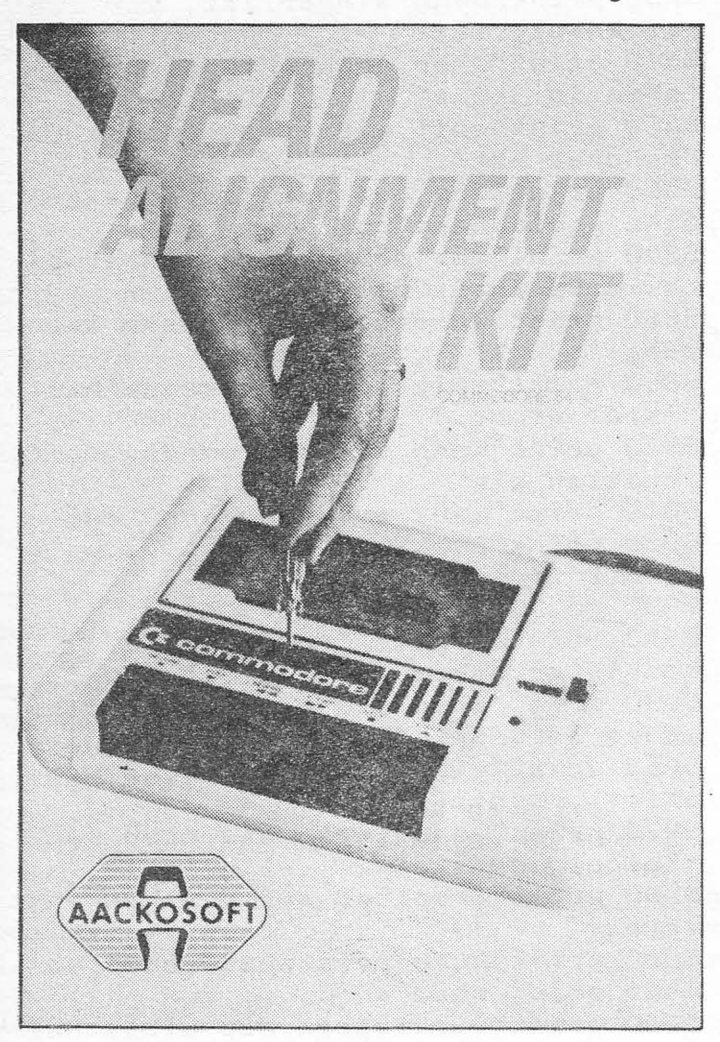
```
4380 e$=e$+ch$:next
4390 fg=a(e):iffg>40-athenfg=40-a
4400 ifa(e)=0then4420
4410 fora=1tofg:e$=e$+rf$+bl$+chr$(
120):next
4420 printho$e$;
4430 fora=ytoy+int((25-y)*.49)
4440 f$=f$+cd$:next
4450 fora=xtox+int((39-x)*.14)
4460 f$=f$+ch$:next
4470 fg=d(e):iffg>40-athenfg=40-a
4480 iffg=0then4500
4490 fora=1tofg:f$=f$+ye$+rf$+chr$(
97):next
4500 printho$f$;
4510 z=s(e)-d(e)*100
4520 g$=g$+cd$
4530 ifz<0thenz=.1
4540 z=z*5/l(e)*10
4550 ifz>10thenz=10
4560 z=10-z
4570 z=int((24-y)*(z/10))+1
4580 fora=1toz
4590 g$=g$+cd$:next
4600 fora=1to(38-x)
4610 g$=g$+ch$:next
4620 printho$g$bl$rn$chr$(95);
4630 printho$cd$cd$wh$tab(14)"jahr:
"y(0)
4640 ifeg=1thenprintho$ye$tab(165)t
$(e)"X"n$(e)
4650 printho$;:gosub520:return
4660 printcl$;:color 0,9,0:color 4,
6,0
4670 printbk$t$(e)" "n$(e)
4680 gosub510
4690 printbl$"staatsinvestitionsmoe
glichkeiten:"cd$
4700 printbk$"(1) markt.........(1)
  1000 goldstuecke"
4710 print"(2) muehle........(1)  2
000 goldstuecke"
4720 print"(3) palast.....(teil)..3
000 goldstuecke"
4730 print"(4) kirche.....(teil)..5
000 goldstuecke"
4740 print"(5) landsknechte (20)...
500 goldstuecke"
4750 printcd$ye$"sie haben"int(k(e)
)"goldstuecke."cd$
4760 printwh$"(0) = fortfahren / (6
) = spielstand"cd$
4770 input"ihre wahl";i
4780 ifi<1thenreturn
4790 ifi>5thengosub1270:goto4660
4800 onigoto4850,4810,4900,4950,500
0
4810 d(e)=d(e)+1
4820 k(e)=k(e)-2000
4830 u(e)=u(e)+.25
4840 goto4660
4850 a(e)=a(e)+1
4860 m(e)=m(e)+5
4870 k(e)=k(e)-1000
4880 u(e)=u(e)+.1
4890 goto4660
4900 b(e)=b(e)+1
4910 n(e)=n(e)+int(rnd(2)*2)+1
4920 k(e)=k(e)-3000
4930 u(e)=u(e)+.5
4940 goto4660
4950 c(e)=c(e)+1
4960 q(e)=q(e)+int(rnd(6)*6)+1
4970 k(e)=k(e)-5000
4980 u(e)=u(e)+1
4990 goto4660
5000 p(e)=p(e)+20
5010 s(e)=s(e)-20
5020 k(e)=k(e)-500
5030 goto4660
5040 printtab(80)
5050 z=0
5060 a=a(e):gosub5300
5070 a=b(e):gosub5300
5080 a=c(e):gosub5300
5090 a=d(e):gosub5300
5100 a=k(e)/5000:gosub5300
5110 a=l(e)/6000:gosub5300
5120 a=m(e)/50:gosub5300
5130 a=n(e)/5:gosub5300
5140 a=p(e)/50:gosub5300
5150 a=q(e)/10:gosub5300
5160 a=s(e)/2000:gosub5300
5170 a=u(e)/5:gosub5300
5180 a=z/u(0)-j(e)
5190 a=int(a)
5200 ifa>8thena=8
5210 if(y(0)+2)=o(e)thent(e)=t(e)+1
5220 ift(e)>=athen5290
5230 t(e)=a
5240 restore
5250 forb=1to(t(e)+v(e))
5260 readt$(e)
5270 next
5280 ift(e)=8then5340
5290 return
5300 ifa>10thena=10
5310 a=int(a)
5320 z=z+a
5330 return
5340 printcl$;:color 0,4,0:color 4,
7,0
5350 eg=1
5360 gosub3800
5370 gosub1270
5380 goto10
```

```
5390 printcl$cy$;:color 0,3,0:color
 4,3,0
5400 printt$(e)" "n$(e)" ist bankro
tt"cd$
5410 printwh$"die glaeubiger haben
sich das meiste "
5420 print"ihres vermoegens angeeig
net."cd$
5430 print"viele leibeigene verlass
en das land"cd$
5440 gosub520
5450 a(e)=0:b(e)=0:c(e)=0:d(e)=0
5460 l(e)=6000:u(e)=1:k(e)=100
5470 s(e)=int(s(e)*rnd(1))
5480 ifr(e)>5000thenr(e)=5000
5490 return
5500 z=0:fora=1tof
5510 ifa=ethen5550
5520 ifp(a)<p(e)then5550
5530 ifp(a)<(1.2*(l(a)/1000))then55
50
5540 ifp(a)>p(z)thenz=a
5550 next
5560 ifz<>0then5600
5570 t$(0)="baron":n$(0)="igor v. f
ronkenstein"
5580 a1=1001+int(rnd(8)*9000)
5590 goto5610
5600 a1=p(z)*1000-l(z)/3
5610 ifa1>(l(e)-5000)thena1=(l(e)-5
000)/2
5620 printre$:forzz=1to40:printz1$;
:nextzz
5630 printbk$t$(z)" "n$(z)" greift
an"
5640 print"und besetzt"a1"morgen la
nd!"
5650 l(z)=l(z)+a1:l(e)=l(e)-a1
5660 z=int(rnd(4)*40)+1:ifz>(p(e)-1
5)thenz=p(e)-15
5670 printt$(e)" "n$(e)
5680 print"verliert"z"soldaten."
5690 printbk$:forzz=1to40:printz1$;
:nextzz
5700 p(e)=p(e)-z
5710 print:gosub520:return
5720 color 0,2,7:color 4,2,7
5730 printcl$bl$" "rn$"...........
.koenig arthus............"cd$bk$
5740 print"sie sind der herrscher e
ines verarmten "
5750 print"geschlechtes aus dem 13.
 jahrhundert sie";
5760 print"muessen durch kauf, verk
auf von land und";
5770 print"weizen, festlegung von s
teuern, zuwei-"
5780 print"sung von weizen, anschaf
fung von guetern";
5790 print"ihr geschick im regieren
 beweisen. je  "
5800 print"besser sie sind, desto s
chneller er-"
5810 print"reichen sie das ziel in
der regierungs-"
5820 print"hierarchie: "bl$"koenig"
bk$" zu werden."
5830 print"denn sieger ist, wer zue
rst koenig ist."
5840 print"aber vorsicht, denn das
leben in jener"
5850 print"zeit war kurz, und schon
 mancher ist"
5860 print"vor dem ziel auf der str
ecke geblieben!"cd$
5870 print"der computer zeichnet ei
ne karte ihres"
5880 print"landes. die groesse der
burg im eck"
5890 print"zeigt die angemessenheit
 ihrer armee."
5900 print"wenn sie schrumpft, dann
 ruesten sie"
5910 print"mehr soldaten aus!"cd$
5920 print" "bk$rn$"...........weit
er mit taste.........."
5930 getb$:ifb$=""then5930
5940 printcl$bl$" "rn$"...........
koenig arthus............"cd$bk$
5950 print"wenn die markierung die
mauer beruehrt,"
5960 print"wird im ganzen land prod
uziert, sonst"
5970 print"benoetigen sie noch mehr
 leibeigene als"
5980 print"arbeiter auf den bauernh
oefen."
5990 print"hoehere steuern bringen
mehr geld, aber"
6000 print"verringern das wirtschaf
tswachstum."
6010 print"sie koennen schulden mac
hen, aber seien"
6020 print"sie vorsichtig, denn die
 zinsen liegen"
6030 print"bei 50% pro jahr und sie
 koennten daran"
6040 print"leicht bankrott gehen!"
6050 printcd$:gosub520:return
6060 rem koenig arthus ==========p4
6070 rem 60671 bytes memory    ====
6080 rem 16084 bytes program   ====
6090 rem 00000 bytes variables ====
6100 rem 00000 bytes arrays    ====
6110 rem 00000 bytes strings   ====
6120 rem 44587 bytes free      ====
```

# EINSTELLHILFE

Das Set besteht aus 2 Cassetten mit Schraubendreher und Anleitung im stabilem Aufbewahrungskasten.

Nie mehr Ladeprobleme, so wird dieses Tonkopfjustage-Set angepriesen. Es ermöglicht die genaue Einstellung des Tonkopfes Ihrer Datasette. Nach dem Laden und Starten des Programmes Head Alignment Kit erscheint eine Graphic, die aus einer großen, vierstelligen Digitalanzeige und einer Skala mit einem rotem und grünem Endbereich besteht, auf der sich ein Zeiger in Abhängigkeit von der Tonkopfeinstellung bewegt. Auf der zweiten Cassette befindet sich ein Prüfton, mit dem der Tonkopf der Datasette eingestellt werden soll. Die 7seitige bebilderte englische Anleitung zeigt ganz genau, wie und wo mit dem beiliegenden Schraubendreher die Einstellung vorgenommen werden muß. Wenn der Tonkopf genau eingestellt ist, sollte sich der Zeiger im grünen Bereich befinden und die Anzeige 3000 oder mehr anzeigen. Die sehr ausführliche Bedienungsanleitung führt Schritt für Schritt durch den Einstell- und Prüfvorgang. Es wird sogar darauf hingewiesen, wann der Schraubendreher aus dem Einstelloch gezogen werden muß.




# ROLLER-BALL

**Für diejenigen, die schon mit dem Monitor umgehen können, folgen jetzt zwei Spiele in Maschinensprache programmiert. Denn die ist nicht nur beim Eintippen, sondern auch beim Ablauf schneller. Die Grafikbalken am Rand müssen Sie nicht mit eingeben, die erzeugt der Computer allein, wenn Sie die Zeile mit Return abschließen! Sie dient der Kontrolle, ob Sie richtig abgetippt haben!**

Vernichten Sie die Monster, die sich in einem Leiterlabyrinth bewegen, indem Sie ihnen Kugeln entgegenrollen. Die Kugeln werden mit der Spacetaste ausgelöst, aber nur in die Richtung, in die Sie sich vorher bewegt haben! Also, wenn Sie nach rechts gelaufen sind, rollt die Kugel nach rechts. Für ein getroffenes Monster gibt es 100 Punkte. Sollte es Ihnen gelingen, den im Labyrinth herumfliegenden Vogel zu erwischen, verwandelt sich dieser in einen Stein, der an einem anderen Ort auftaucht. Dafür erhalten Sie dann 200 Punkte.
Ihre Bewegungen im Labyrinth werden mit den Cursortasten in alle Richtungen gesteuert. Das Spiel hat 15 Durchgänge, wobei nach jedem dritten Durchgang das Labyrinth verändert wird und mit neuen und gefährlicheren Monstern aufwartet.
Wurden Sie von einem Monster, Vogel oder Stein erwischt, bzw. wenn die Zeit abgelaufen ist, haben Sie verloren.
Da das Spiel ganz in Maschinensprache geschrieben ist, geben Sie es bitte mit Hilfe des eingebauten Monitors ein und saven es vor dem Starten auf Kassette oder Diskette ab. SAVE"Rollerball",1,1408,2F92 für Kassette oder SAVE"Rollerball",8,1408,2F92 für Diskette.)

Geladen wird das Spiel mit LOAD"Rollerball",1,1 bzw. mit LOAD "Rollerball",8,1. Gestartet wird das Spiel mit SYS 4988. Und nun viel Erfolg bei der Monsterjagd.

# WALLBREAK

Dieses Spiel ist dem Klassiker "Break Out" nachempfunden und kann als eine gelungene Variante für den C-16 bezeichnet werden. Sie müssen versuchen, mit dem Schläger am unterem Bildschirmrand den Ball so zurückzuschlagen, daß Sie alle Steine aus den am oberen Bildrand befindlichen Rhomben herausschlagen. Der Schläger wird mit den Cursor-Tasten bewegt. Da das Spiel in Maschinensprache geschrieben ist, geben Sie es bitte mit dem eingebauten Monitor ein und saven es vor dem Start auf Kassette oder Diskette ab. (SAVE "Wall-Break",1,1038,1AA7 für Kassette oder mit SAVE"Wall-Break",8,1038,1AA7 für Diskette)
Geladen wird das Spiel mit LOAD"Wall-Break",1,1 oder alternativ mit ,8,1. Gestartet wird das Spiel mit SYS 5320, nachdem Sie mit X >RETURN< aus dem Monitor ausgestiegen sind.

```
; FB000 00 00 00 00 F8

>01408 7E 7E 7E 7E 00 7E FF FF FF FF FF FF E7 E7 E0 E0:
>01418 FF FF FF FF E7 E7 07 07 FF FF FF FF E7 E7 E7 E7:
>01428 00 00 00 FF FF 00 00 00 00 3C 7E 7E 7E 7E 3C 00:
>01438 E3 77 3D 1D 0F 03 01 00 C7 EE BC B8 F0 C0 80 00:
>01448 03 07 05 0D 1F 3B 71 E0 C0 E0 A0 B0 F8 DC 8E 07:
>01458 01 03 06 07 07 07 03 01 80 C0 E0 60 A0 E0 C0 80:
>01468 00 03 07 0F 0E 05 03 00 00 C0 E0 70 F0 E0 C0 00:
>01478 13 03 0F 12 05 3A 30 08 09 13 03 0F 12 05 3A 30:
>01488 20 00 07 20 01 20 0D 20 05 20 20 0F 20 16 20 05:
>01498 20 12 00 10 12 05 13 13 20 13 10 01 03 05 00 12:
>014A8 20 0F 20 0C 20 0C 20 05 20 12 20 20 02 20 01 20:
>014B8 0C 20 0C 00 14 09 0D 05 3A 40 40 40 40 40 40 40:
>014C8 A2 00 BD 00 D0 9D 00 10 BD 00 D1 9D 00 11 BD 00:
>014D8 D2 9D 00 12 BD 00 D3 9D 00 13 E8 D0 E5 A9 05 20:
>014E8 D2 FF A9 00 8D 15 FF 8D 19 FF A9 C0 8D 12 FF A9:
>014F8 10 8D 13 FF 4C 0C 1F EA 00 20 20 20 20 20 20 20:
>01508 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20:
>01518 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 00:
>01528 00 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20:
>01538 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20:
>01548 20 20 20 20 20 20 20 00 00 20 20 6B 25 73 20 20:
>01558 20 20 20 20 20 20 20 20 20 20 6B 25 73 20 20 20:
>01568 20 20 20 20 20 20 20 20 20 20 6B 25 73 20 20 00:
>01578 00 70 40 6B 25 73 40 40 40 40 40 40 40 40 40 40:
>01588 40 40 6B 25 73 40 40 40 40 40 40 40 40 40 40 40:
>01598 40 40 6B 25 73 40 6E 00 00 6D 40 6B 25 73 40 40:
>015A8 40 40 40 40 40 40 40 40 40 40 6B 25 73 40 40 40:
>015B8 40 40 40 40 40 40 40 40 40 40 6B 25 73 40 7D 00:
>015C8 00 20 20 6B 25 73 20 20 20 20 20 20 20 20 20 20:
>015D8 20 20 6B 25 73 20 20 20 20 20 20 20 20 20 20 20:
>015E8 20 20 6B 25 73 20 20 00 00 20 20 6B 25 73 20 20:
>015F8 20 20 20 6B 25 73 20 20 20 20 6B 25 73 20 20 20:
>01608 20 20 6B 25 73 20 20 20 20 20 6B 25 73 20 20 00:
>01618 00 70 40 40 40 40 40 40 40 40 40 6B 25 73 40 40:
>01628 40 40 40 40 40 40 40 40 40 40 6B 25 73 40 40 40:
>01638 40 40 6B 25 73 40 6E 00 00 6D 40 40 40 40 40 40:
>01648 40 40 40 6B 25 73 40 40 40 40 40 40 40 40 40 40:
>01658 40 40 6B 25 73 40 40 40 40 40 6B 25 73 40 7D 00:
>01668 00 20 20 20 20 20 20 20 20 20 20 6B 25 73 20 20:
>01678 20 20 20 20 20 20 20 20 20 20 6B 25 73 20 20 20:
>01688 20 20 6B 25 73 20 20 00 00 20 20 6B 25 73 20 20:
>01698 20 20 20 6B 25 73 20 20 20 20 20 20 20 20 20 20:
>016A8 20 20 6B 25 73 20 20 20 20 20 6B 25 73 20 20 00:
>016B8 00 70 40 6B 25 73 40 40 40 40 40 6B 25 73 40 40:
>016C8 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40:
>016D8 40 40 6B 25 73 40 6E 00 00 6D 40 6B 25 73 40 40:
>016E8 40 40 40 6B 25 73 40 40 40 40 40 40 40 40 40 40:
>016F8 40 40 40 40 40 40 40 40 40 40 6B 25 73 40 7D 00:
>01708 00 20 20 6B 25 73 20 20 20 20 20 6B 25 73 20 20:
>01718 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20:
>01728 20 20 6B 25 73 20 20 00 00 20 20 6B 25 73 20 20:
>01738 20 20 20 6B 25 73 20 20 20 20 6B 25 73 20 20 20:
>01748 20 20 20 20 20 20 20 20 20 20 6B 25 73 20 20 00:
>01758 00 70 40 6B 25 73 40 40 40 40 40 40 40 40 40 40:
>01768 40 40 40 6B 25 73 40 40 40 40 40 40 40 40 40 40:
>01778 40 40 6B 25 73 40 6E 00 00 6D 40 6B 25 73 40 40:
>01788 40 40 40 40 40 40 40 40 40 40 40 6B 25 73 40 40:
>01798 40 40 40 40 40 40 40 40 40 40 6B 25 73 40 7D 00:
>017A8 00 20 20 6B 25 73 20 20 20 20 20 20 20 20 20 20:
```

```
>017B8 20 20 20 6B 25 73 20 20 20 20 20 20 20 20 20 20:
>017C8 20 20 6B 25 73 20 20 00 00 20 20 6B 25 73 20 20:
>017D8 20 20 20 20 20 20 20 20 20 20 20 6B 25 73 20 20:
>017E8 20 20 20 20 20 20 20 20 20 20 6B 25 73 20 20 00:
>017F8 00 70 40 40 40 40 40 40 40 40 40 40 40 40 40 40:
>01808 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40:
>01818 40 40 40 40 40 40 6E 00 20 6D 40 40 40 40 40 40:
>01828 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40:
>01838 40 40 40 40 40 40 40 40 40 40 40 40 40 40 7D 00:
>01848 A2 00 BD A7 14 9D 48 0D A9 45 9D 48 09 E8 E0 14:
>01858 D0 F0 A2 00 BD 9B 14 9D EC 0D A9 49 9D EC 09 E8:
>01868 E0 0B D0 F0 A9 00 8D 33 03 8D 34 03 8D 46 03 8D:
>01878 45 03 A5 C6 C9 3C D0 FA 20 6B C5 A2 00 BD 08 14:
>01888 9D 08 11 E8 E0 70 D0 F5 A9 00 8D 35 03 8D 36 03:
>01898 A9 64 8D 37 03 A9 00 8D 38 03 A9 32 8D F9 19 A9:
>018A8 14 8D 39 03 A9 24 85 E0 A9 23 85 E1 A9 01 85 E2:
>018B8 A9 15 85 D7 A9 00 8D 3A 03 A9 04 85 DB A9 06 85:
>018C8 D8 A9 00 85 E3 A9 01 85 E4 A0 00 B9 7F 14 99 01:
>018D8 0C C8 C0 08 D0 F5 A2 00 BD 78 14 9D 1C 0C E8 E0:
>018E8 07 D0 F5 A2 00 BD BC 14 9D 71 0F A9 49 9D 71 08:
>018F8 E8 E0 05 D0 F0 EA A2 00 A9 6C 9D 78 0F A9 55 9D:
>01908 78 0B E8 E0 1E D0 F1 EA EA A9 02 85 DE 85 DF A2:
>01918 00 A9 0E 9D 00 3F A9 19 9D 0D 3F E8 E0 0C D0 F1:
>01928 A2 00 AD 00 FF C9 01 90 F9 C9 03 B0 F5 9D 19 3F:
>01938 E8 E0 0C D0 ED 4C A5 1D 8A 48 A9 0C 85 D1 A9 08:
>01948 85 D3 A2 00 A9 00 85 D0 85 D2 18 65 D4 B0 0C E8:
>01958 E0 28 D0 F7 85 D0 85 D2 68 AA 60 18 E6 D1 E6 D3:
>01968 90 ED EA EA 8A 48 A2 00 A9 15 85 D6 A9 00 85 D5:
>01978 18 65 D4 B0 0A E8 E0 28 D0 F7 85 D5 68 AA 60 18:
>01988 E6 D6 90 F1 EA 98 48 A2 01 A0 01 86 D4 20 40 19:
>01998 20 6C 19 B1 D5 91 D0 A9 6D 91 D2 C8 C0 27 D0 F3:
>019A8 E8 E0 15 D0 E4 68 A8 60 8A 48 A2 00 A9 FF 38 E9:
>019B8 01 D0 FC E8 EC 3B 03 D0 F3 68 AA 60 A4 DE A6 DF:
>019C8 86 D4 20 40 19 A5 F7 91 D0 A9 55 91 D2 C8 A5 F8:
>019D8 91 D0 A9 55 91 D2 A5 F7 C9 2B F0 09 A9 2B 85 F7:
>019E8 A9 2C 85 F8 60 A9 2D 85 F7 A9 2E 85 F8 60 EA EA:
>019F8 A9 32 8D 3B 03 20 B0 19 60 A2 00 BD 19 3F C9 01:
>01A08 F0 12 C9 02 F0 2E C9 04 F0 49 C9 03 F0 63 E8 E4:
>01A18 DB D0 E8 60 BC 0D 3F BD 00 3F 85 D4 20 40 19 C8:
>01A28 C8 B1 D0 C9 73 F0 05 FE 0D 3F D0 E2 20 F3 1A FE:
>01A38 0D 3F D0 DA BC 0D 3F BD 00 3F 85 D4 20 40 19 88:
>01A48 B1 D0 C9 6B F0 05 DE 0D 3F D0 C3 20 F3 1A DE 0D:
>01A58 3F D0 BB BC 0D 3F FE 00 3F BD 00 3F 85 D4 20 40:
>01A68 19 B1 D0 C9 40 F0 02 D0 A5 20 F3 1A DE 00 3F D0:
>01A78 9D BC 0D 3F DE 00 3F BD 00 3F 85 D4 20 40 19 B1:
>01A88 D0 C9 20 F0 02 D0 87 20 F3 1A FE 00 3F 4C 16 1A:
>01A98 A2 00 BD 19 3F C9 00 F0 23 BC 0D 3F BD 00 3F 85:
>01AA8 D4 20 40 19 A5 E7 91 D0 A9 77 91 D2 C8 A5 E8 91:
>01AB8 D0 A9 77 91 D2 86 E5 4C D2 1D A6 E5 E8 E4 DB D0:
>01AC8 D1 A5 E7 C5 03 F0 09 A5 03 85 E7 A5 04 85 E8 60:
>01AD8 A5 05 85 E7 A5 06 85 E8 60 AD 35 03 18 6D 37 03:
>01AE8 B0 04 20 F7 1E 60 EE 36 03 D0 F7 AD 00 FF C9 01:
>01AF8 90 F9 C9 05 B0 F5 9D 19 3F 60 A2 00 BC 0D 3F C0:
>01B08 25 F0 0A C0 01 F0 0D E8 E4 DB D0 F0 60 A9 02 9D:
>01B18 19 3F D0 F3 A9 01 9D 19 3F D0 EC EA A6 E2 E0 01:
>01B28 F0 0D E0 02 F0 24 E0 03 F0 39 E0 04 F0 4C 60 A4:
>01B38 DE A6 DF 86 D4 20 40 19 C8 C8 B1 D0 C9 73 F0 03:
>01B48 E6 DE 60 20 AE 1B E6 DE 60 EA A4 DE A6 DF 86 D4:
>01B58 20 40 19 88 B1 D0 C9 6B F0 03 C6 DE 60 20 AE 1B:
>01B68 C6 DE 60 A4 DE A6 DF CA 86 D4 20 40 19 B1 D0 C9:
>01B78 20 F0 03 C6 DF 60 20 AE 1B 60 A4 DE A6 DF E8 86:
```

```
>01B88 D4 20 40 19 B1 D0 C9 40 F0 03 E6 DF 60 20 AE 1B:
>01B98 60 A4 DE C0 25 F0 05 C0 01 F0 06 60 A9 02 85 E2:
>01BA8 60 A9 01 85 E2 60 AD 00 FF C9 01 90 F9 C9 05 B0:
>01BB8 F5 85 E2 60 EA A5 C6 C9 33 F0 11 C9 30 F0 3E C9:
>01BC8 3C F0 6B A9 24 85 E1 A9 22 85 E1 60 A4 D7 C0 25:
>01BD8 F0 2A A6 D8 86 D4 20 40 19 C8 C8 B1 D0 C9 40 F0:
>01BE8 1B CA 86 D4 20 40 19 B1 D0 C9 40 F0 0F E6 D7 A9:
>01BF8 01 8D 3A 03 A9 22 85 F9 A9 44 1C E1 60 A4 D7 C0:
>01C08 02 F0 2A A6 D8 86 D4 20 40 19 88 88 B1 D0 C9 40:
>01C18 F0 1B CA 86 D4 20 40 19 B1 D0 C9 40 F0 0F C6 D7:
>01C28 A9 02 8D 3A 03 A9 22 85 F9 A9 23 85 E1 60 AD 3A:
>01C38 03 C9 01 F0 05 C9 02 F0 11 60 A9 00 54 1C 1B A9:
>01C48 01 85 E3 84 D9 A5 D8 85 DA 60 A9 00 F2 1C 1B A9:
>01C58 02 85 E3 84 D9 A5 D8 85 DA 60 A5 C6 C9 2B F0 05:
>01C68 C9 28 F0 22 60 A6 D8 E0 02 F0 1A A4 D7 CA CA 86:
>01C78 D4 20 40 19 B1 D0 C9 40 B0 0B C6 D8 A9 22 85 F9:
>01C88 A9 03 8D 3A 03 60 A6 D8 A4 D7 E8 86 D4 20 40 19:
>01C98 B1 D0 C9 40 B0 17 C9 6E F0 13 C9 70 F0 0F E6 D8:
>01CA8 A9 22 85 F9 A9 23 85 E1 A9 03 8D 3A 03 60 A4 D7:
>01CB8 A6 D8 86 D4 CA 86 D4 20 40 19 A9 21 91 D0 A9 6F:
>01CC8 91 D2 E8 86 D4 20 40 19 A5 E0 91 D0 A9 6F 91 D2:
>01CD8 A5 F9 C9 20 F0 0A A5 E0 C9 22 F0 05 A9 22 85 E0:
>01CE8 60 A9 23 85 E0 60 EA EA A2 00 13 1D 8D 11 FF A9:
>01CF8 07 8D 0E FF A9 32 8D 3B 03 20 B0 19 A9 C0 8D 0E:
>01D08 FF 20 B0 19 E8 E0 05 D0 E6 29 00 A1 1D FF 60 A4:
>01D18 D7 A6 D8 86 D4 20 40 19 B1 D0 C9 55 B0 2A C9 27:
>01D28 90 26 A4 D7 A6 D8 86 D4 20 40 19 A9 56 91 D0 A9:
>01D38 45 91 D2 C6 D4 20 40 19 A9 56 91 D0 A9 45 91 D2:
>01D48 20 F0 1C 68 68 4C 6F 1E CA 86 D4 20 40 19 B1 D0:
>01D58 C9 55 B0 07 C9 27 90 03 4C 2A 1D 60 EA A5 E3 C9:
>01D68 01 F0 05 C9 02 F0 05 60 E6 D9 D0 02 C6 D9 A4 D9:
>01D78 A6 DA 86 D4 20 40 19 A9 26 91 D0 A9 57 91 D2 A9:
>01D88 01 8D 3B 03 20 6F 1D A5 D9 C9 26 B0 05 C9 02 90:
>01D98 01 60 A9 3C 8D C8 1B A9 00 EE 1D 1F EA 20 8C 19:
>01DA8 20 65 1D 20 01 1A 20 98 1A 20 02 1B 20 24 1B 20:
>01DB8 C4 19 20 99 1B 20 17 1D 20 BD 1B 20 62 1C 20 B6:
>01DC8 1C 20 2E 1E 20 F8 19 4C 8E 21 A4 D9 A6 DA 86 D4:
>01DD8 20 40 19 B1 D2 C9 77 F0 04 4C C2 1A EA A9 3C 8D:
>01DE8 C8 1B A6 E5 A9 00 29 1E 85 D9 85 E3 9D 19 3F 20:
>01DF8 E1 1A 20 0D 1E A5 DB C5 E4 F0 05 E6 E4 4C C2 1A:
>01E08 68 68 4C 95 1F A9 1F 8D 11 FF A9 A0 8D 0E FF A9:
>01E18 14 8D 3B 03 20 B0 19 A9 C0 8D 0E FF 20 B0 19 29:
>01E28 00 6B 1E FF 60 EA AD 39 03 38 E9 01 F0 04 8D 39:
>01E38 03 60 A9 20 AE 38 03 9D 78 0F EE 38 03 E0 1D F0:
>01E48 09 A9 14 8D 39 03 20 57 1E 60 68 68 4C 24 1F A9:
>01E58 1F 8D 11 FF A9 A9 8D 0E FF A9 28 8D 3B 03 20 B0:
>01E68 19 29 00 71 1E FF 60 A2 00 83 1E 14 9D 4A 0D A9:
>01E78 CA 9D 4A 09 E8 E0 10 D0 F0 A2 00 9B 1E 14 9D 3D:
>01E88 0E A9 64 9D 3D 0A E8 E0 08 D0 F0 A5 C6 C9 3C D0:
>01E98 FA A2 00 AC 1E 9D 22 0C E8 E0 06 D0 F8 A9 3C 8D:
>01EA8 C8 1B A9 00 CA 1E 85 D9 AD 34 03 CD 36 03 90 04:
>01EB8 F0 2A B0 0C AD 36 03 8D 34 03 AD 35 03 8D 33 03:
>01EC8 A2 00 DB 1E 18 20 F0 FF AD 34 03 AE 33 03 20 5F:
>01ED8 A4 A9 00 FC 1E 03 8D 36 03 4C 72 21 AD 33 03 CD:
>01EE8 35 03 90 02 B0 DA AD 35 03 8D 33 03 4C C8 1E 8D:
>01EF8 35 03 A2 00 64 1F 18 20 F0 FF AD 36 03 AE 35 03:
>01F08 20 5F A4 60 A9 3C 8D C8 1B A9 41 85 E7 A9 42 85:
>01F18 E8 A9 27 85 F7 A9 28 85 F8 4C C9 20 A4 D7 A6 D8:
>01F28 86 D4 20 40 19 A9 24 91 D0 A9 6F 91 D2 C6 D4 20:
>01F38 40 19 A9 21 91 D0 A9 6F 91 D2 20 F8 19 E6 D8 A5:
>01F48 D8 C9 17 F0 20 A9 1F 8D 11 FF A9 FF 8D 0E FF 20:
```

```
>01F58 F8 19 A9 0F 8D 0E FF 20 F8 19 29 00 6F 1F FF 20:
>01F68 8C 19 4C 24 1F A9 00 BE 1F FF A4 D7 C6 D8 A6 D8:
>01F78 86 D4 20 40 19 A9 20 91 D0 C6 D4 20 40 19 A9 20:
>01F88 91 D0 4C 6F 1E 85 E3 85 DA 85 D9 60 EA 20 EB 1F:
>01F98 A5 DB C9 0C D0 03 20 E9 20 A5 DB 18 69 04 C9 10:
>01FA8 D0 02 A9 04 85 DB A9 14 85 D7 A9 06 85 D8 A9 0E:
>01FB8 85 DF 85 DE A2 00 D3 1F FF C9 01 90 F9 C9 03 B0:
>01FC8 F5 9D 19 3F E8 E0 0C D0 ED A2 00 0B 20 9D 78 0F:
>01FD8 A9 55 9D 78 0B E8 E0 1E D0 F1 20 F0 1C 4C A5 1D:
>01FE8 EA EA EA A2 0F A0 1F 8C 11 FF 8E 0E FF A9 64 8D:
>01FF8 3B 03 20 B0 19 88 C0 10 F0 07 8A 18 69 0A AA D0:
>02008 E6 A9 00 15 20 FF A9 00 8D 38 03 A2 00 1E 20 9D:
>02018 0D 3F A9 0E 9D 00 8A 20 E0 0C D0 F1 A9 01 85 E4:
>02028 60 03 3F 7D CD 9F 3F 63 C0 C0 FC BE B3 FD FC C6:
>02038 03 C3 7F 3D 0D 1F 3F 33 18 C3 FE BC B0 F8 FC CC:
>02048 18 83 C7 ED 7D 3F 0E 05 03 C0 E0 B0 BC FE 77 A3:
>02058 C1 03 07 0D 3D 7F EE C5 83 C1 E3 B7 BE FC 70 A0:
>02068 C0 0F 1F 33 33 3F 3F 3B 31 F0 F8 CC CC FC FC DC:
>02078 8C 0F 1F 3F 3F 3F 3F 3B 31 F0 F8 FC FC FC FC DC:
>02088 8C 00 A2 20 03 05 0D 1F 3F 1F FE F8 F0 A0 B0 F8:
>02098 FC F8 FE 1F 0F 05 0D 1F 5D 00 CB 20 C0 A0 B0 F8:
>020A8 FC 1F 1F 19 1F 1F 18 0C 06 F8 F8 98 F8 F8 18 30:
>020B8 60 1F 1F 19 1F 1F 18 30 60 F8 F8 98 F8 F8 18 0C:
>020C8 06 A2 00 6E 21 20 9D 08 12 E8 E0 A0 D0 F5 A9 41:
>020D8 85 03 A9 42 85 04 A9 43 85 05 A9 44 85 06 4C 48:
>020E8 18 EE 45 03 AD 45 03 C9 01 F0 11 C9 02 F0 22 C9:
>020F8 03 F0 33 C9 04 F0 44 C9 05 F0 55 60 A9 45 85 E7:
>02108 85 03 A9 46 85 E8 85 04 A9 47 85 05 A9 48 85 06:
>02118 60 A9 49 85 E7 85 03 A9 4A 85 E8 85 04 A9 4B 85:
>02128 05 A9 4C 85 06 60 A9 4D 85 E7 85 03 A9 4E 85 E8:
>02138 85 04 A9 4F 85 05 A9 50 85 06 60 A9 51 85 E7 85:
>02148 03 A9 52 85 E8 85 04 A9 53 85 05 A9 54 85 06 60:
>02158 A9 41 85 E7 85 03 A9 42 85 E8 85 04 A9 43 85 05:
>02168 A9 44 85 06 A9 00 88 21 03 60 A9 41 85 E8 85 03:
>02178 A9 42 85 E8 85 04 A9 43 85 05 A9 44 85 06 A9 00:
>02188 BE 21 03 4C 90 18 A4 D9 A6 DA 86 D4 20 40 19 B1:
>02198 D0 C9 2F B0 2C C9 27 90 28 20 0D 1E 20 43 2F A9:
>021A8 C8 8D 37 03 20 E1 1A A9 64 8D 37 03 A9 0E 85 DF:
>021B8 A9 04 85 DE A9 00 DE 21 85 DA 85 D9 A9 3C 8D C8:
>021C8 18 AD 45 03 C9 01 F0 13 C9 02 F0 1C C9 03 F0 25:
>021D8 C9 04 F0 2E C9 00 1E 22 4C A5 1D A9 25 8D 71 19:
>021E8 A9 6B 8D 75 19 4C A5 1D A9 22 8D 71 19 A9 24 8D:
>021F8 75 19 4C A5 1D A9 28 8D 71 19 A9 B3 8D 75 19 4C:
>02208 A5 1D A9 2B 8D 71 19 A9 FB 8D 75 19 4C A5 1D A9:
>02218 15 8D 71 19 A9 00 25 22 19 4C A5 1D 00 4C 22 20:
>02228 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20:
>02238 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20:
>02248 20 20 20 00 74 22 20 20 20 20 20 20 20 20 20 20:
>02258 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20:
>02268 20 20 20 20 20 20 20 20 20 20 20 00 9C 22 20 6B:
>02278 25 73 20 20 20 20 20 20 20 20 20 20 20 20 20 20:
>02288 20 20 20 20 20 20 20 20 20 20 20 20 20 20 6B 25:
>02298 73 20 20 00 C4 22 40 6B 25 73 40 40 40 40 40 40:
>022A8 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40:
>022B8 40 40 40 40 40 40 6B 25 73 40 6E 00 EC 22 40 6B:
>022C8 25 73 40 40 40 40 40 40 40 40 40 40 40 40 40 40:
>022D8 40 40 40 40 40 40 40 40 40 40 40 40 40 40 6B 25:
>022E8 73 40 7D 00 14 23 20 6B 25 73 20 20 20 20 20 20:
>022F8 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20:
>02308 20 20 20 20 20 20 6B 25 73 20 20 00 3C 23 20 6B:
>02318 25 73 20 20 20 20 6B 25 73 20 20 20 20 20 20 20:
```

# LISTINGS

```
>02328 20 20 20 20 20 20 6B 25 73 20 20 20 20 20 6B 25:
>02338 73 20 20 00 64 23 40 40 40 40 40 40 40 40 6B 25:
>02348 73 40 40 40 40 40 40 40 40 40 40 40 40 40 6B 25:
>02358 73 40 40 40 40 40 40 40 40 40 6E 00 8C 23 40 40:
>02368 40 40 40 40 40 40 6B 25 73 40 40 40 40 40 40 40:
>02378 40 40 40 40 40 40 6B 25 73 40 40 40 40 40 40 40:
>02388 40 40 7D 00 B4 23 20 20 20 20 20 20 20 20 6B 25:
>02398 73 20 20 20 20 20 20 20 20 20 20 20 20 20 6B 25:
>023A8 73 20 20 20 20 20 20 20 20 20 20 00 DC 23 20 6B:
>023B8 25 73 20 20 20 20 6B 25 73 20 20 20 20 20 6B 25:
>023C8 73 20 20 20 20 20 6B 25 73 20 20 20 20 20 6B 25:
>023D8 73 20 20 00 04 24 40 6B 25 73 40 40 40 40 40 40:
>023E8 40 40 40 40 40 40 6B 25 73 40 40 40 40 40 40 40:
>023F8 40 40 40 40 40 40 6B 25 73 40 6E 00 2C 24 40 6B:
>02408 25 73 40 40 40 40 40 40 40 40 40 40 40 40 6B 25:
>02418 73 40 40 40 40 40 40 40 40 40 40 40 40 40 6B 25:
>02428 73 40 7D 00 54 24 20 6B 25 73 20 20 20 20 20 20:
>02438 20 20 20 20 20 20 6B 25 73 20 20 20 20 20 20 20:
>02448 20 20 20 20 20 20 6B 25 73 20 20 00 7C 24 20 6B:
>02458 25 73 20 20 20 20 20 20 20 20 20 20 20 20 6B 25:
>02468 73 20 20 20 20 20 20 20 20 20 20 20 20 20 6B 25:
>02478 73 20 20 00 A4 24 40 6B 25 73 40 40 40 40 40 40:
>02488 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40:
>02498 40 40 40 40 40 40 6B 25 73 40 6E 00 CC 24 40 6B:
>024A8 25 73 40 40 40 40 40 40 40 40 40 40 40 40 40 40:
>024B8 40 40 40 40 40 40 40 40 40 40 40 40 40 40 6B 25:
>024C8 73 40 7D 00 F4 24 20 6B 25 73 20 20 20 20 20 20:
>024D8 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20:
>024E8 20 20 20 20 20 20 6B 25 73 20 20 00 1C 25 20 6B:
>024F8 25 73 20 20 20 20 20 20 20 20 20 20 20 20 20 20:
>02508 20 20 20 20 20 20 20 20 20 20 20 20 20 20 6B 25:
>02518 73 20 20 00 44 25 40 40 40 40 40 40 40 40 40 40:
>02528 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40:
>02538 40 40 40 40 40 40 40 40 40 40 6E 00 6C 25 40 40:
>02548 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40:
>02558 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40:
>02568 40 40 7D 00 93 25 20 20 20 20 20 20 20 20 20 20:
>02578 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20:
>02588 20 20 20 20 20 20 20 20 20 20 00 BB 25 20 20 20:
>02598 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20:
>025A8 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20:
>025B8 20 20 00 E3 25 20 6B 25 73 20 20 20 20 20 20 20:
>025C8 20 20 6B 25 73 20 20 20 20 20 20 20 20 20 20 20:
>025D8 20 20 20 20 20 6B 25 73 20 20 00 0B 26 40 6B 25:
>025E8 73 40 40 40 40 40 40 40 40 40 6B 25 73 40 40 40:
>025F8 40 40 40 40 40 40 40 40 40 40 40 40 40 6B 25 73:
>02608 40 6E 00 33 26 40 6B 25 73 40 40 40 40 40 40 40:
>02618 40 40 6B 25 73 40 40 40 40 40 40 40 40 40 40 40:
>02628 40 40 40 40 40 6B 25 73 40 7D 00 5B 26 20 6B 25:
>02638 73 20 20 20 20 20 20 20 20 20 6B 25 73 20 20 20:
>02648 20 20 20 20 20 20 20 20 20 20 20 20 6B 25 73:
>02658 20 20 00 83 26 20 6B 25 73 20 20 20 20 20 20 20:
>02668 20 20 6B 25 73 20 20 20 20 20 20 20 6B 25 73:
>02678 20 20 20 20 20 6B 25 73 20 20 00 AB 26 40 6B 25:
>02688 73 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40:
>02698 40 40 40 40 40 6B 25 73 40 40 40 40 40 40 40 40:
>026A8 40 6E 00 D3 26 40 6B 25 73 40 40 40 40 40 40 40:
>026B8 40 40 40 40 40 40 40 40 40 40 40 40 40 6B 25 73:
>026C8 40 40 40 40 40 40 40 40 40 7D 00 FB 26 20 6B 25:
>026D8 73 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20:
>026E8 20 20 20 20 20 6B 25 73 20 20 20 20 20 20 20 20:
```

```
>026F8 20 20 00 23 27 20 6B 25 73 20 20 20 20 20 20 20:
>02708 20 20 20 20 20 6B 25 73 20 20 20 20 20 6B 25 73:
>02718 20 20 20 20 20 6B 25 73 20 20 00 4B 27 40 40 40:
>02728 40 40 40 40 40 40 40 40 40 40 40 40 40 6B 25 73:
>02738 40 40 40 40 40 40 40 40 40 40 40 40 40 6B 25 73:
>02748 40 6E 00 73 27 40 40 40 40 40 40 40 40 40 40 40:
>02758 40 40 40 40 40 6B 25 73 40 40 40 40 40 40 40 40:
>02768 40 40 40 40 40 6B 25 73 40 7D 00 9B 27 20 20 20:
>02778 20 20 20 20 20 20 20 20 20 20 20 20 20 6B 25 73:
>02788 20 20 20 20 20 20 20 20 20 20 20 20 20 6B 25 73:
>02798 20 20 00 C3 27 20 6B 25 73 20 20 20 20 20 20 20:
>027A8 20 20 20 20 20 6B 25 73 20 20 20 20 20 20 20 20:
>027B8 20 20 20 20 20 6B 25 73 20 20 00 EB 27 40 6B 25:
>027C8 73 40 40 40 40 40 40 40 40 40 40 40 40 6B 25 73:
>027D8 40 40 40 40 40 40 40 40 40 40 40 40 40 6B 25 73:
>027E8 40 6E 00 13 28 40 6B 25 73 40 40 40 40 40 40 40:
>027F8 40 40 40 40 40 6B 25 73 40 40 40 40 40 40 40 40:
>02808 40 40 40 40 40 6B 25 73 40 7D 00 3B 28 20 6B 25:
>02818 73 20 20 20 20 20 20 20 20 20 20 20 20 6B 25 73:
>02828 20 20 20 20 20 20 20 20 20 20 20 20 20 6B 25 73:
>02838 20 20 00 63 28 20 6B 25 73 20 20 20 20 20 20 20:
>02848 20 20 20 20 20 6B 25 73 20 20 20 20 20 20 20 20:
>02858 20 20 20 20 20 6B 25 73 20 20 00 8B 28 40 40 40:
>02868 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40:
>02878 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40:
>02888 40 6E 00 B3 28 40 40 40 40 40 40 40 40 40 40 40:
>02898 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40:
>028A8 40 40 40 40 40 40 40 40 40 7D 00 DB 28 20 20 20:
>028B8 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20:
>028C8 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20:
>028D8 20 20 00 03 29 20 20 20 20 20 20 20 20 20 20 20:
>028E8 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20:
>028F8 20 20 20 20 20 20 20 20 20 20 00 2B 29 20 6B 25:
>02908 73 20 20 20 20 20 20 20 20 20 20 20 6B 25 73 20:
>02918 20 20 20 20 20 20 20 20 20 20 20 20 20 6B 25 73:
>02928 20 20 00 53 29 40 6B 25 73 40 40 40 40 40 40 40:
>02938 40 40 40 40 6B 25 73 40 40 40 40 40 40 40 40 40:
>02948 40 40 40 40 40 6B 25 73 40 6E 00 7B 29 40 6B 25:
>02958 73 40 40 40 40 40 40 40 40 40 40 40 6B 25 73 40:
>02968 40 40 40 40 40 40 40 40 40 40 40 40 40 6B 25 73:
>02978 40 7D 00 A3 29 20 6B 25 73 20 20 20 20 20 20 20:
>02988 20 20 20 20 6B 25 73 20 20 20 20 20 20 20 20 20:
>02998 20 20 20 20 20 6B 25 73 20 20 00 CB 29 20 6B 25:
>029A8 73 20 20 20 20 20 20 20 20 20 20 20 6B 25 73 20:
>029B8 20 20 20 20 20 6B 25 73 20 20 20 20 20 6B 25 73:
>029C8 20 20 00 F3 29 40 6B 25 73 40 40 40 40 40 40 40:
>029D8 40 40 40 40 40 40 40 40 40 40 40 40 40 6B 25 73:
>029E8 40 40 40 40 40 40 40 40 40 6E 00 1B 2A 40 6B 25:
>029F8 73 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40:
>02A08 40 40 40 40 40 6B 25 73 40 40 40 40 40 40 40 40:
>02A18 40 7D 00 43 2A 20 6B 25 73 20 20 20 20 20 20 20:
>02A28 20 20 20 20 20 20 20 20 20 20 20 20 20 6B 25 73:
>02A38 20 20 20 20 20 20 20 20 20 20 00 6B 2A 20 6B 25:
>02A48 73 20 20 20 20 20 6B 25 73 20 20 20 20 20 20 20:
>02A58 20 20 20 20 20 6B 25 73 20 20 20 20 20 6B 25 73:
>02A68 20 20 00 93 2A 40 6B 25 73 40 40 40 40 40 6B 25:
>02A78 73 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40:
>02A88 40 40 40 40 40 6B 25 73 40 6E 00 BB 2A 40 6B 25:
>02A98 73 40 40 40 40 40 6B 25 73 40 40 40 40 40 40 40:
>02AA8 40 40 40 40 40 40 40 40 40 40 40 40 40 6B 25 73:
>02AB8 40 7D 00 E3 2A 20 6B 25 73 20 20 20 20 20 6B 25:
```

```
>02AC8 73 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20:
>02AD8 20 20 20 20 20 6B 25 73 20 20 00 0B 2B 20 6B 25:
>02AE8 73 20 20 20 20 20 6B 25 73 20 20 20 20 20 6B 25:
>02AF8 73 20 20 20 20 20 20 20 20 20 20 20 20 6B 25 73:
>02B08 20 20 00 33 2B 40 6B 25 73 40 40 40 40 40 40 40:
>02B18 40 40 40 40 40 40 6B 25 73 40 40 40 40 40 40 40:
>02B28 40 40 40 40 40 6B 25 73 40 6E 00 5B 2B 40 6B 25:
>02B38 73 40 40 40 40 40 40 40 40 40 40 40 40 40 6B 25:
>02B48 73 40 40 40 40 40 40 40 40 40 40 40 40 6B 25 73:
>02B58 40 7D 00 83 2B 20 6B 25 73 20 20 20 20 20 20 20:
>02B68 20 20 20 20 20 20 6B 25 73 20 20 20 20 20 20 20:
>02B78 20 20 20 20 20 6B 25 73 20 20 00 AB 2B 20 6B 25:
>02B88 73 20 20 20 20 20 20 20 20 20 20 20 20 20 6B 25:
>02B98 73 20 20 20 20 20 20 20 20 20 20 20 20 6B 25 73:
>02BA8 20 20 00 D3 2B 40 40 40 40 40 40 40 40 40 40 40:
>02BB8 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40:
>02BC8 40 40 40 40 40 40 40 40 40 6E 00 FB 2B 40 40 40:
>02BD8 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40:
>02BE8 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40:
>02BF8 40 7D 00 23 2C 20 20 20 20 20 20 20 20 20 20 20:
>02C08 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20:
>02C18 20 20 20 20 20 20 20 20 20 20 00 4B 2C 20 20 20:
>02C28 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20:
>02C38 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20:
>02C48 20 20 00 73 2C 20 6B 25 73 20 20 20 20 20 20 20:
>02C58 20 20 20 20 20 6B 25 73 20 20 20 20 20 20 20 20:
>02C68 20 20 20 20 20 6B 25 73 20 20 00 9B 2C 40 6B 25:
>02C78 73 40 40 40 40 40 40 40 40 40 40 40 40 6B 25 73:
>02C88 40 40 40 40 40 40 40 40 40 40 40 40 40 6B 25 73:
>02C98 40 6E 00 C3 2C 40 6B 25 73 40 40 40 40 40 40 40:
>02CA8 40 40 40 40 40 6B 25 73 40 40 40 40 40 40 40 40:
>02CB8 40 40 40 40 40 6B 25 73 40 7D 00 EB 2C 20 6B 25:
>02CC8 73 20 20 20 20 20 20 20 20 20 20 20 20 6B 25 73:
>02CD8 20 20 20 20 20 20 20 20 20 20 20 20 20 6B 25 73:
>02CE8 20 20 00 13 2D 20 6B 25 73 20 20 20 20 20 6B 25:
>02CF8 73 20 20 20 20 6B 25 73 20 20 20 20 20 6B 25 73:
>02D08 20 20 20 20 20 6B 25 73 20 20 00 3B 2D 40 40 40:
>02D18 40 40 40 40 40 40 6B 25 73 40 40 40 40 40 40 40:
>02D28 40 40 40 40 40 6B 25 73 40 40 40 40 40 6B 25 73:
>02D38 40 6E 00 63 2D 40 40 40 40 40 40 40 40 40 6B 25:
>02D48 73 40 40 40 40 40 40 40 40 40 40 40 40 6B 25 73:
>02D58 40 40 40 40 40 6B 25 73 40 7D 00 8B 2D 20 20 20:
>02D68 20 20 20 20 20 20 6B 25 73 20 20 20 20 20 20 20:
>02D78 20 20 20 20 20 6B 25 73 20 20 20 20 20 6B 25 73:
>02D88 20 20 00 B3 2D 20 6B 25 73 20 20 20 20 20 6B 25:
>02D98 73 20 20 20 20 20 20 20 20 20 20 20 20 6B 25 73:
>02DA8 20 20 20 20 20 6B 25 73 20 20 00 DB 2D 40 6B 25:
>02DB8 73 40 40 40 40 40 6B 25 73 40 40 40 40 40 40 40:
>02DC8 40 40 40 40 40 40 40 40 40 40 40 40 40 6B 25 73:
>02DD8 40 6E 00 03 2E 40 6B 25 73 40 40 40 40 40 6B 25:
>02DE8 73 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40:
>02DF8 40 40 40 40 40 6B 25 73 40 7D 00 2B 2E 20 6B 25:
>02E08 73 20 20 20 20 20 6B 25 73 20 20 20 20 20 20 20:
>02E18 20 20 20 20 20 20 20 20 20 20 20 20 20 6B 25 73:
>02E28 20 20 00 53 2E 20 6B 25 73 20 20 20 20 20 6B 25:
>02E38 73 20 20 20 20 20 6B 25 73 20 20 20 20 20 20 20:
>02E48 20 20 20 20 20 6B 25 73 20 20 00 7B 2E 40 6B 25:
>02E58 73 40 40 40 40 40 40 40 40 40 40 40 40 40 6B 25:
>02E68 73 40 40 40 40 40 40 40 40 40 40 40 40 6B 25 73:
>02E78 40 6E 00 A3 2E 40 6B 25 73 40 40 40 40 40 40 40:
>02E88 40 40 40 40 40 40 6B 25 73 40 40 40 40 40 40 40:
```

```
>02E98 40 40 40 40 40 6B 25 73 40 7D 00 CB 2E 20 6B 25:
>02EA8 73 20 20 20 20 20 20 20 20 20 20 20 20 20 6B 25:
>02EB8 73 20 20 20 20 20 20 20 20 20 20 20 20 6B 25 73:
>02EC8 20 20 00 F3 2E 20 6B 25 73 20 20 20 20 20 20 20:
>02ED8 20 20 20 20 20 20 6B 25 73 20 20 20 20 20 20 20:
>02EE8 20 20 20 20 20 6B 25 73 20 20 00 1B 2F 40 40 40:
>02EF8 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40:
>02F08 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40:
>02F18 40 6E 00 43 2F 40 40 40 40 40 40 40 40 40 40 40:
>02F28 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40:
>02F38 40 40 40 40 40 40 40 40 40 7D 00 8D 2F 03 C9 00:
>02F48 F0 05 C9 01 F0 22 60 A9 2B 85 F7 8D E1 19 8D E5:
>02F58 19 A9 2C 8D E9 19 85 F8 A9 2D 8D EE 19 A9 2E 8D:
>02F68 F2 19 A9 01 8D 46 03 60 A9 27 85 F7 8D E1 19 8D:
>02F78 E5 19 A9 28 85 F8 8D E9 19 A9 29 8D EE 19 A9 2A:
>02F88 8D F2 19 A9 00 C1 2F 03 60 EA FF FF FF FF FF FF:
```

# WALLBREAK

```
; FB000 00 00 00 00 F8

>01030 4E 0D 4C 49 53 54 0D 4D 20 20 20 64 65 20 20 20:
>01040 20 20 64 65 64 65 20 20 20 64 65 64 65 64 65 20:
>01050 64 65 64 65 64 65 64 65 20 64 65 64 65 64 65 20:
>01060 20 20 64 65 64 65 20 20 20 20 20 64 65 20 20 20:
>01070 64 65 64 65 64 65 64 65 20 64 65 64 65 64 65 20:
>01080 20 20 64 65 64 65 20 20 20 20 20 64 65 20 20 20:
>01090 20 20 64 65 64 65 20 20 20 64 65 64 65 64 65 20:
>010A0 64 65 64 65 64 65 64 65 64 65 64 65 64 65 64 65:
>010B0 20 64 65 64 65 64 65 20 64 65 64 65 64 65 64 65:
>010C0 20 64 65 64 65 64 65 20 64 65 64 65 64 65 64 65:
>010D0 20 64 65 64 65 64 65 20 64 65 64 65 64 65 64 65:
>010E0 64 65 64 65 64 65 64 65 64 65 64 65 64 65 64 65:
>010F0 64 65 64 65 64 65 64 65 64 65 64 65 64 65 64 65:
>01100 64 65 64 65 64 65 64 65 64 65 64 65 64 65 64 65:
>01110 64 65 64 65 64 65 64 65 20 20 20 64 65 20 20 20:
>01120 20 20 64 65 64 65 20 20 20 64 65 64 65 64 65 20:
>01130 20 20 20 64 65 20 20 20 20 20 64 65 64 65 20 20:
>01140 20 64 65 64 65 64 65 20 64 65 64 65 64 65 64 65:
>01150 20 20 20 64 65 20 20 20 20 20 64 65 64 65 20 20:
>01160 20 64 65 64 65 64 65 20 64 65 64 65 64 65 64 65:
>01170 64 65 64 65 64 65 64 65 64 65 64 65 64 65 00 65:
>01180 64 65 64 65 64 65 64 65 20 00 64 65 64 65 20 20:
>01190 20 64 65 64 00 64 65 20 64 65 64 65 64 65 64 00:
>011A0 64 65 64 65 64 65 64 65 64 65 00 65 64 65 64 65:
>011B0 20 64 65 64 65 00 65 20 20 20 64 65 64 65 20 20:
>011C0 00 65 20 20 20 20 20 20 64 65 64 00 20 20 20 20:
>011D0 64 65 64 65 64 65 20 20 64 65 64 65 64 65 64 65:
>011E0 64 65 64 65 64 65 20 20 64 7E 02 65 20 20 20 20:
>011F0 64 65 20 20 20 20 20 20 64 65 64 65 64 65 64 65:
>01200 64 65 64 00 20 2C 2E 24 FF FF FF 65 FF 65 64 65:
>01210 A7 1A 64 65 64 65 64 65 20 64 65 64 65 64 65 20:
>01220 20 20 10 65 FF 65 FF 20 FF 20 01 04 65 20 20 00:
>01230 41 41 41 65 64 65 20 20 20 20 64 65 64 65 20 09:
>01240 C0 00 05 94 09 09 09 89 09 00 C0 F0 50 40 21 00:
>01250 90 41 00 41 21 11 81 11 21 41 41 41 11 65 20 20:
>01260 20 20 64 65 64 65 20 06 00 00 00 00 00 02 08 02:
>01270 00 64 65 64 0F 0F 65 20 20 20 64 65 64 65 20 20:
>01280 20 20 20 64 65 FF FF FF 20 20 64 65 64 65 20 20:
```

```
>01290 20 64 65 64 65 64 65 20 64 65 64 65 64 65 64 65:
>012A0 64 65 64 65 64 65 64 65 20 64 65 20 20 64 65 20:
>012B0 20 64 65 64 65 64 65 20 20 64 65 20 20 64 65 20:
>012C0 20 64 65 64 65 64 65 20 20 64 65 20 20 64 65 20:
>012D0 64 65 64 65 64 65 64 65 00 3C 42 7E 7E 42 3C 00:
>012E0 FF FF FF 00 00 00 00 00 FF FF FF 00 00 00 00 00:
>012F0 FF FF FF 00 00 00 00 00 00 FF FF FF FF 00 FF 00:
>01300 00 FE FE FE FE FE FE 00 3F 3F 3F 00 00 00 00 00:
>01310 FF FF FF 00 00 00 00 00 FC FC FC 00 00 00 00 00:
>01320 C0 FC BE EF E5 30 F8 54 FF C3 A5 99 99 A5 C3 FF:
>01330 53 43 52 45 45 4E 3A 31 20 20 20 20 20 20 20 66:
>01340 67 68 66 67 68 20 20 20 20 53 43 4F 52 45 3A 30:
>01350 30 00 47 20 41 20 4D 20 45 20 20 4F 20 56 20 45:
>01360 20 52 00 50 52 45 53 53 20 53 50 41 43 45 00 65:
>01370 57 41 4C 4C 2D 42 52 45 41 4B 65 00 A9 1C 20 D2:
>01380 FF A9 93 20 D2 FF A2 04 A0 0E 18 20 F0 FF A0 00:
>01390 B9 D8 10 C8 C0 0D F0 05 20 4C FF D0 F3 A2 06 A0:
>013A0 0E 18 20 F0 FF A0 00 B9 D8 10 C8 C0 0D F0 05 20:
>013B0 4C FF D0 F3 A2 05 A0 0E 18 20 F0 FF A0 00 B9 6F:
>013C0 13 C9 00 F0 06 20 4C FF C8 D0 F3 A9 82 20 D2 FF:
>013D0 A9 05 20 D2 FF A2 0B A0 0E 18 20 F0 FF A0 00 B9:
>013E0 63 13 C9 00 F0 06 20 4C FF C8 D0 F3 A9 84 20 D2:
>013F0 FF A9 00 8D 15 FF 8D 19 FF A9 02 8D 12 FF A9 30:
>01400 8D 13 FF A2 00 BD 00 D0 9D 00 30 BD 00 D1 9D 00:
>01410 31 E8 D0 F1 A2 00 BD D8 12 9D 00 32 E8 E0 58 D0:
>01420 F5 A5 C6 C9 3C D0 FA A9 00 8D 33 03 8D 34 03 A9:
>01430 19 8D 35 03 A9 16 8D 36 03 A9 00 8D 37 03 A9 4C:
>01440 8D 47 03 A9 38 85 D6 A9 10 85 D7 A9 1E 8D 38 03:
>01450 A9 13 8D 3F 03 8D 40 03 A9 01 8D 42 03 A9 04 8D:
>01460 48 03 A9 00 8D 44 03 EA A9 93 20 D2 FF A2 00 A0:
>01470 02 18 20 F0 FF A0 00 B9 30 13 C9 00 F0 06 20 4C:
>01480 FF C8 D0 F3 A2 00 A9 7F 9D 0E 08 E8 E0 09 D0 F8:
>01490 A9 1C 20 D2 FF A9 04 8D 3C 03 A5 D6 8D 39 03 A5:
>014A0 D7 8D 3A 03 A2 04 AC 3C 03 18 20 F0 FF A0 00 B1:
>014B0 D6 20 4C FF C8 C0 08 D0 F6 A5 D6 18 69 08 B0 20:
>014C0 85 D6 E8 E0 0B D0 DF AD 39 03 85 D6 AD 3A 03 85:
>014D0 D7 AD 3C 03 18 69 0C C9 28 B0 0A 8D 3C 03 90 C4:
>014E0 18 E6 D7 90 DB 4C D5 17 A2 00 BD A0 0C C9 44 F0:
>014F0 11 BD 40 0D C9 44 F0 0A E8 E0 A0 D0 ED 68 68 4C:
>01500 88 16 60 18 AD 33 03 6D 35 03 B0 05 8D 33 03 90:
>01510 06 18 EE 34 03 90 F5 A2 00 A0 21 18 20 F0 FF A9:
>01520 05 20 D2 FF AD 34 03 AE 33 03 20 5F A4 60 EA A9:
>01530 82 20 D2 FF A9 81 20 D2 FF A2 0C A0 0C 18 20 F0:
>01540 FF A0 00 B9 52 13 C9 00 F0 06 20 4C FF C8 D0 F3:
>01550 A9 84 20 D2 FF A9 05 20 D2 FF A2 0E A0 0E 18 20:
>01560 F0 FF A0 00 B9 63 13 C9 00 F0 06 20 4C FF C8 D0:
>01570 F3 A5 C6 C9 3C D0 FA 4C 27 14 A9 0C 85 D3 A9 08:
>01580 85 D5 A2 00 A9 00 85 D2 85 D4 18 6D 3D 03 B0 0A:
>01590 E8 E0 28 D0 F6 85 D2 85 D4 60 18 E6 D3 E6 D5 90:
>015A0 EF EA A5 D6 18 69 38 B0 1E 85 D6 A5 D7 C9 12 F0:
>015B0 03 4C 95 14 A5 D6 C9 D8 F0 02 D0 F5 A9 38 85 D6:
>015C0 A9 10 85 D7 4C 95 14 18 E6 D7 90 DD EA A2 FF CA:
>015D0 D0 FD EE 44 03 AD 44 03 CD 3B 03 D0 F0 A9 00 8D:
>015E0 44 03 60 A2 01 A9 4A 9D 48 0F A9 5F 9D 48 0B E8:
>015F0 E0 06 D0 F1 A2 22 A9 4A 9D 48 0F A9 5F 9D 48 0B:
>01600 E8 E0 27 D0 F1 60 68 68 4C 45 18 AC 36 03 A9 20:
>01610 99 00 0C 88 99 00 0C 88 99 00 0C 88 8C 36 03 C0:
>01620 0D F0 03 4C E2 19 4C 2F 15 EA AE 40 03 8E 41 03:
>01630 A5 C6 C9 33 F0 2B C9 30 F0 39 AE 40 03 8E 41 03:
>01640 A9 41 9D 70 0F A9 7F 9D 70 0B E8 A9 42 9D 70 0F:
>01650 A9 7F 9D 70 0B E8 A9 43 9D 70 0F A9 7F 9D 70 0B:
```

```
>01660 60 E0 25 F0 FB E8 8E 40 03 AE 41 03 A9 20 9D 70:
>01670 0F D0 C7 E0 00 F0 E9 CA 8E 40 03 AE 41 03 A9 20:
>01680 E8 E8 9D 70 0F D0 B3 EA A9 05 20 D2 FF AD 42 03:
>01690 69 01 8D 42 03 C9 0D F0 18 A2 00 A0 09 18 20 F0:
>016A0 FF A9 00 AE 42 03 20 5F A4 A9 1C 20 D2 FF 4C 86:
>016B0 1A A2 00 A0 0A 18 20 F0 FF A9 20 20 4C FF A9 01:
>016C0 8D 42 03 A0 09 18 20 F0 FF A9 00 AE 42 03 20 5F:
>016D0 A4 A9 1C 20 D2 FF 4C 86 1A EA AD 3B 03 48 A9 19:
>016E0 8D 3B 03 A9 2F 8D 11 FF A9 C8 8D 0F FF 20 CD 15:
>016F0 A9 00 8D 11 FF 68 8D 3B 03 60 EA AD 3B 03 48 A9:
>01700 14 8D 3B 03 A9 1F 8D 11 FF A9 FF 8D 0E FF 20 CD:
>01710 15 A9 00 8D 11 FF 68 8D 3B 03 60 EA AD 33 03 CD:
>01720 47 03 B0 01 60 AD 34 03 CD 48 03 B0 01 60 18 AD:
>01730 47 03 69 E8 B0 10 8D 47 03 AD 48 03 18 69 03 8D:
>01740 48 03 20 E3 15 60 EE 48 03 B0 EB 00 AC 37 03 EE:
>01750 3D 03 20 7A 15 C8 B1 D2 C9 4A F0 0C 88 88 B1 D2:
>01760 C9 4A F0 04 CE 3D 03 60 20 DA 16 CE 3D 03 A9 20:
>01770 91 D2 4C 06 16 EA AD 3D 03 8D 3D 03 20 7A 15 AC:
>01780 37 03 C8 B1 D2 C9 45 F0 09 88 88 B1 D2 C9 44 F0:
>01790 33 60 A9 20 91 D2 88 91 D2 20 03 15 20 7B 1A 68:
>017A0 68 AD 38 03 C9 01 D0 03 4C 2F 19 C9 02 D0 03 4C:
>017B0 7F 19 C9 03 D0 03 4C 8F 18 C9 04 D0 03 4C DF 18:
>017C0 C9 05 F0 E4 A9 20 91 D2 C8 91 D2 20 03 15 20 7B:
>017D0 1A 4C 9F 17 EA EA 20 DA 16 20 DA 16 20 DA 16 A0:
>017E0 14 8C 37 03 A9 0B 8D 3D 03 20 7A 15 A9 40 91 D2:
>017F0 A9 64 91 D4 20 CD 15 20 2A 16 AC 37 03 EE 3D 03:
>01800 20 7A 15 B1 D2 C9 41 F0 22 C9 42 F0 1E C9 43 F0:
>01810 1A AD 3D 03 C9 17 F0 19 CE 3D 03 20 7A 15 A9 20:
>01820 91 D2 EE 3D 03 20 FB 16 4C E9 17 CE 3D 03 4C 45:
>01830 18 CE 3D 03 20 7A 15 A9 20 91 D2 20 DA 16 20 FB:
>01840 16 4C CF 19 EA 20 DA 16 A9 05 8D 38 03 AC 37 03:
>01850 20 7A 15 A9 20 91 D2 CE 3D 03 20 7A 15 A9 40 91:
>01860 D2 A9 64 91 D4 20 CD 15 20 2A 16 20 1C 17 20 76:
>01870 17 20 4C 17 20 FD 19 AC 37 03 AD 3D 03 C9 01 F0:
>01880 03 4C 4D 18 C0 14 90 03 4C 7F 19 4C 2F 19 EA 20:
>01890 DA 16 A9 01 8D 38 03 AC 37 03 20 7A 15 A9 20 91:
>018A0 D2 EE 37 03 AC 37 03 CE 3D 03 20 7A 15 A9 40 91:
>018B0 D2 A9 64 91 D4 20 CD 15 20 2A 16 20 1C 17 20 76:
>018C0 17 20 4C 17 20 FD 19 AC 37 03 C0 27 F0 0D AD 3D:
>018D0 03 C9 03 90 03 4C 97 18 4C 2F 19 4C DF 18 EA 20:
>018E0 DA 16 A9 02 8D 38 03 AC 37 03 20 7A 15 A9 20 91:
>018F0 D2 CE 37 03 AC 37 03 CE 3D 03 20 7A 15 A9 40 91:
>01900 D2 A9 64 91 D4 20 CD 15 20 2A 16 20 1C 17 20 76:
>01910 17 20 4C 17 20 FD 19 AC 37 03 C0 00 F0 0D AD 3D:
>01920 03 C9 03 90 03 4C E7 18 4C 7F 19 4C 8F 18 EA 20:
>01930 DA 16 A9 03 8D 38 03 AC 37 03 20 7A 15 A9 20 91:
>01940 D2 EE 37 03 AC 37 03 EE 3D 03 20 7A 15 A9 40 91:
>01950 D2 A9 64 91 D4 20 CD 15 20 2A 16 20 1C 17 20 76:
>01960 17 20 4C 17 20 FD 19 AC 37 03 C0 27 F0 0D AD 3D:
>01970 03 C9 17 B0 03 4C 37 19 4C CF 19 4C 7F 19 EA 20:
>01980 DA 16 A9 04 8D 38 03 AC 37 03 20 7A 15 A9 20 91:
>01990 D2 CE 37 03 AC 37 03 EE 3D 03 20 7A 15 A9 40 91:
>019A0 D2 A9 64 91 D4 20 CD 15 20 2A 16 20 1C 17 20 76:
>019B0 17 20 4C 17 20 FD 19 AC 37 03 C0 00 F0 0D AD 3D:
>019C0 03 C9 17 B0 03 4C 87 19 4C CF 19 4C 2F 19 EA AC:
>019D0 37 03 20 7A 15 A9 20 91 D2 20 FB 16 20 DA 16 4C:
>019E0 0B 16 AE 40 03 A9 20 9D 70 0F E8 9D 70 0F E8 9D:
>019F0 70 0F A2 13 8E 40 03 4C D5 17 EA EA EA AC 37 03:
>01A00 EE 3D 03 20 7A 15 C8 B1 D2 AC 37 03 C9 41 F0 32:
>01A10 88 B1 D2 AC 37 03 C9 43 F0 4A AC 37 03 B1 D2 C9:
>01A20 41 F0 1F C9 42 F0 2C C9 43 F0 39 CE 3D 03 20 7A:
```

```
>01A30 15 AD 3D 03 C9 17 B0 3D 20 E8 14 20 2A 16 20 CD:
>01A40 15 60 68 68 CE 3D 03 20 E8 14 20 2A 16 20 CD 15:
>01A50 4C 8F 18 68 68 CE 3D 03 20 E8 14 20 2A 16 20 CD:
>01A60 15 4C 45 18 68 68 CE 3D 03 20 E8 14 20 2A 16 20:
>01A70 CD 15 4C DF 18 68 68 4C CF 19 EA 20 DA 16 20 FB:
>01A80 16 20 DA 16 60 EA AE 40 03 A9 20 9D 70 0F E8 9D:
>01A90 70 0F E8 9D 70 0F A2 13 8E 40 03 AC 37 03 A9 20:
>01AA0 91 D2 4C A2 15 FF FF FF FF FF FF FF FF FF FF FF:
>01AB0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF:
```

# MIXED DATEI 16

Als erstes erfolgt die Auswahl zwischen einer reinen Textdatei oder einer gemischten Zahlen/Textdatei.
Nach Erscheinen des Menüs können folgende Unterprogramme aufgerufen werden:
Neue Datei eingeben
Speichern einer Datei
Bestehende Datei laden
Anfügen an bestehende Datei
Daten auf Bildschirm
Print (Daten auf Drucker)
Löschen der Daten
Umschalten zur Zahlen-Text-Datei
Ende
Nach Aufruf des Modus >Neueingabe< erfolgt die Abfrage >Dateiname<. Nach Ende Daten speichern, falls nötig Datendisk einlegen.
Menüpunkt > Speichern einer Datei<: Sollte es Probleme beim Abspeichern der Datei geben, wird automatisch eine Disk-Fehlermeldung in Klartext gedruckt. Menüpunkt > Daten laden<: Nach Abfrage des Dateinamens erfolgt entweder eine Übernahmebestätigung oder eine Disk-Fehlermeldung im Klartext.
Bei der allgemeinen Textdatei sind Komma, Semikolon, Doppelpunkt, Anführungsstriche zur Eingabe erlaubt.
Menüpunkt > Anfügen an eine bestehende Datei<: Sind keine Daten im Speicher, informiert der Computer darüber. Es wird bei bestehenden Daten die Anzahl angezeigt, um weitere Daten anzufügen. Eingabe mit> * < beenden. Danach Abfrage >Daten speichern?<, falls nötig, Datendisk einlegen.
Menüpunkt >Daten auf Bildschirm<: Abfrage >Einzelschritte per Tastendruck oder angezeigt, Eingabe beenden mit >*<
Menüpunkt >Drucken<: Nach Aufruf dieses Programmteils ist zur Druckeranpassung die Änderung der Sekundäradresse möglich. Drucker einschalten nicht vergessen, sonst Programmabsturz!
Menüpunkt >Löschen der Daten<: Nach einer Sicherheitsabfrage sind dann die Daten unwiderbringlich futsch und der Arbeitsspeicher geleert.
Menüpunkt > Umschalten zur Zahlen und Textdatei< (oder umgekehrt): Nach Anzeigen des Zwischenmenüs Text- und Zahlendatei sind folgende Auswahlpunkte möglich:
Neueingabe
File laden
File speichern
File anzeigen
File addieren
File erweitern
Druckerroutine
Ende
Umschalten auf Textdatei



```
10 rem mixed datei =============c16
20 rem (p)         commodore-welt bw-
30 rem ==============================
40 rem (c)          by              =
50 rem harald beiler                =
60 rem                              =
70 rem version 3.5 40z/ascii        =
80 rem c16/p4 + 1530/1541/drucker =
90 rem ==============================

100 color
0,1:color4,1:color1,14:scnclr:print
chr$(14)chr$(11)chr$(27)chr$(69)
110 char1,4,5,"T = reine Textdatei.
..........",1
120 char1,4,7,"Z = Text mit Zahlen
+ Addition",1
130 char1,0,10,"Bitte druecken Sie
entsprechende Taste!":print
140 getkeya$
150 ifa$<>"t"anda$<>"z"then140
160 ifa$="t"thengosub180:goto240
170 ifa$="z"thengosub180:goto1890
180 lz$="
                 ":rem 39 leerzeichen
190 dz$="###,###.##"
200 ze=1000:dima$(ze):n=1000:dimn(n
):return
210 rem ******************************
220 rem ****************** titel *
230 rem ******************************
240 gosub1830:char1,8,3,"Allgemeine
 Text Datei",1:print:gosub660
250 char1,2,6,"Eingabe:
Dateiname oder Nummer":print:char1,
11,7,"die Daten":print:gosub660
260 char1,2,10,"(Alle Eingaben auss
er Anfuehrungs-":print
270 char1,2,11,"zeichen sind erlaub
t !)":print:gosub660
280 char1,12,18,"M = M E N U E",1:p
rint
290 getkeya$:ifa$="m"then300:else:g
oto290
300 scnclr:gosub690:gosub1130
310 getkeyq$
320 ifq$="n"goto450
330 ifq$="h"goto1230
340 ifq$="b"goto900
350 ifq$="d"goto960
360 ifq$="a"goto1560
370 ifq$="e"thenend
380 ifq$="l"goto1750
390 ifq$="s"goto800
400 ifq$="u"goto1890
410 goto310
420 rem ******************************
430 rem ************** neueingabe *
440 rem ******************************
450 scnclr
460 char1,15,2,"NEUEINGABE",1:print
:gosub660:gosub680
470 printchr$(17)tab(2)"Dateiname:
";:poke19,4:inputcm$:poke19,0
480 gosub490:goto730
490 print:ze=1:
500 printtab(2);:poke1319,34:poke23
9,1:inputa$(ze):printchr$(13);:tl=z
e/15:fori=1to40
510 iftl=ithengosub580:printchr$(19
)"Seite: "tl+1:char1,2,8,""
520 next
530 ifa$(ze)=chr$(42)thenreturn
540 ze=ze+1:poke239,0
550 goto500
560 rem ******************************
570 rem ********** unterprogramme *
580 rem ******************************
590 printchr$(18);:char1,2,22,"Seit
e vor=^":char1,20,22,"Menue = ö"
600 getkeya$
610 ifa$=chr$(145)then640
620 ifa$=chr$(92)then300
630 getkeya$
640 char1,0,8,"",0:i=0: dountili=15
:printlz$:i=i+1:loop
650 char1,0,8,"":return
660 printtab(1):i=0:dountili=36:pri
ntchr$(96);:i=i+1:loop:print
670 return
680 char1,7,5,"(Eingabe beeenden mi
t '*')":print:return
690 char1,15,2,"M E N U E",1:print:
gosub660:return
700 i=1:dountili=ze
710 ifa$(i)=""then1700:loop
720 return
730 char1,8,22,"Daten speichern ? (
j/n)",1:print
740 printtab(2)chr$(18)"Falls noeti
g, Datendisk einlegen !"chr$(146)
750 getkeya$
760 ifa$="j"then800:else:goto300
770 rem ***************************
780 rem ******** daten speichern *
790 rem ***************************
800 scnclr:char1,0,2,"Daten speiche
rn",1:print
810 print"File-Name oder -nummer: "
;cm$
820 print:printcm$" wird jetzt gesi
chert":print"Bitte warten.":print
830 open2,8,2,"$0:"+cm$+",s,w"
840 print#2,ze
850 i=1:dountili=ze:print#2,chr$(34
)+a$(i)+chr$(34):i=i+1:loop:close2
```

```
860 gosub1650:goto1030
870 rem ****************************
880 rem ************ daten laden *
890 rem ****************************
900 scnclr:char1,0,2,"Daten laden."
,1:print
910 print"Dateiname: ";:poke19,4:in
putcm$:poke19,0:print
920 open2,8,2,"$0:"+cm$+",s,r":prin
tcm$
930 input#2,ze
940 i=1:dountili=ze:input#2,a$(i):i
=i+1:loop:close2
950 gosub1650:goto300
960 scnclr:char1,0,2,"Bildschirm-Au
sgabe",1:print
970 printtab(2)chr$(18)" E "chr$(14
6)"inzelschritt per Tastendruck"
980 printtab(2)chr$(18)" G "chr$(14
6)"esamte Datei anzeigen":gosub660
990 getkeya$
1000 ifa$="e"then1360
1010 ifa$="g"then1450
1020 else:goto990
1030 printtab(2)chr$(17)chr$(18)" D
 = Drucken "tab(26)" M =
Menue"chr$(146)
1040 getq$
1050 ifq$="d"thengosub1230:goto300
1060 ifq$="m"then300
1070 goto1040
1080 goto1040
1090 else:goto1040
1100 rem ****************************
1110 rem ******* ein/ausgabemenue *
1120 rem ****************************
1130 printchr$(17)tab(2)chr$(18)" N
 "chr$(146)"eue Datei eingeben":pri
nt
1140 printtab(2)chr$(18)" S "chr$(1
46)"peichern einer Datei":print
1150 printtab(2)chr$(18)" B "chr$(1
46)"estehende Datei laden":print
1160 printtab(2)chr$(18)" A "chr$(1
46)"nfuegen an bestehende Datei":pr
int
1170 printtab(2)chr$(18)" D "chr$(1
46)"aten auf Bildschirm":print
1180 printtab(2)chr$(18)" H "chr$(1
46)"ardcopy (Druckerausgabe)":print
1190 printtab(2)chr$(18)" L "chr$(1
46)"oeschen von Daten":print
1200 printtab(2)chr$(18)" U "chr$(1
46)"mschalten zur Zahlen/Textdatei"
:gosub660
1210 printtab(2)chr$(18)" E "chr$(1
46)"NDE":gosub660:return
1220 rem ****************************
1230 rem ******* daten ausdrucken *
1240 rem ****************************
1250 scnclr:char1,0,2,"Drucker-Rout
ine.",1:print
1260 print:print" Aenderung der "
1270 print" Sekundaeradresse"
1280 printtab(19)chr$(145)"7"chr$(1
57)chr$(157)chr$(157);:inputsa
1290 printchr$(17)chr$(18)"Wenn Dru
cker bereit ist, Taste !"chr$(146)
1300 poke239,0:wait239,1
1310 open4,4,sa:cmd4
1320 print#4,cm$:print#4:i=1:dounti
li=ze:print#4,a$(i):i=i+1:loop:prin
t#4:close4:goto300
1330 rem ****************************
1340 rem ***** bildschirm-ausgabe *
1350 rem ****************************
1360 gosub700
1370 printcm$:print:ze=1
1380 printtab(2)a$(ze):tl=ze/15:pok
e239,0:wait239,1
1390 forj=1to40
1400 iftl=jthengosub640
1410 next
1420 ifa$(ze)=chr$(42)then1030
1430 ze=ze+1
1440 goto1380
1450 gosub700
1460 printcm$:print:ze=1
1470 printtab(2)a$(ze):tl=ze/15:for
j=1to40
1480 iftl-jthenprintchr$(18)"Seite
vor=Taste."chr$(146):poke239,0:wait
239,1:gosub640
1490 next
1500 ifa$(ze)=chr$(42)then1030
1510 ze=ze+1
1520 goto1470
1530 rem ****************************
1540 rem append - daten anhaengen *
1550 rem ****************************
1560 scnclr:gosub700:gosub1620:ze=z
e-1
1570 gosub680:gosub660:printtab(2)c
m$:print:printtab(2)a$(ze)
1580 ze=ze+1:printtab(2);:poke1319,
34:poke239,1:poke19,4:inputa$(ze):p
oke19,0:print
1590 ifa$(ze)=chr$(42)then1610
1600 goto1580
1610 print:goto730
1620 ifze=900then1700
1630 printtab(2)chr$(18)"Datensaetz
e bisher:"str$(ze-1)chr$(146)
1640 printtab(2)" Letzter Datensatz
 wird angezeigt.":return
1650 rem ****************************
```

```
1660 rem ************ disk-fehler *
1670 rem ****************************
1680 ifds<>0thenprintds$;ds:cm$="":
goto300
1690 return
1700 gosub1830:printchr$(19)chr$(17
)tab(3)chr$(18)" Keine Daten im Spe
icher !!!"chr$(146)
1710 char1,25,18,"M = MENUE",1:prin
t:goto290
1720 rem ****************************
1730 rem ********* daten loeschen *
1740 rem ****************************
1750 gosub1830:print:printchr$(19)t
ab(2)chr$(17)chr$(18)"Daten loesche
n."chr$(146)
1760 print:printtab(2)"Sind Sie sic
her (j/n) ?":gosub660
1770 getkeya$
1780 ifa$="j"then1790:else:goto300
1790 printtab(2)"Bitte warten...":i
=1:dountili=ze:a$(i)="":i=i+1:loop:
cm$=""
1800 print:printtab(2)chr$(18)" Arb
eitsspeicher ist jetzt leer."chr$(1
46)
1810 fort=1to200:next:goto300
1820 char1,0,10,"":i=1:dountili=10:
printlz$;:loop:return
1830 scnclr
1840 printchr$(176);:i=0
1850 dountili=36:printchr$(96);:i=i
+1:loop:printchr$(174):j=0
1860 dountilj=20:printchr$(125)tab(
37)chr$(125):j=j+1:loop:k=0
1870 printchr$(173);:dountilk=36:pr
intchr$(96);:k=k+1:loop:printchr$(1
89)
1880 return
1890 gosub1830:gosub690
1900 print:printtab(10)chr$(18)"TEX
T/ZAHLENDATEI"chr$(146):print
1910 printtab(9)" 1 = Neueingabe"
1920 printtab(9)" 2 = File laden"
1930 printtab(9)" 3 = File speicher
n"
1940 printtab(9)" 4 = File anzeigen
"
1950 printtab(9)" 5 = File addieren
"
1960 printtab(9)" 6 = File erweiter
n"
1970 printtab(9)" 7 = Drucker-Routi
ne"
1980 printtab(9)" 8 = ENDE
1990 print:printtab(9)" 9 = Umschal
ten auf "
2000 printtab(14)"Textdatei"
2010 getq$:ifq$=""then2010
2020 onval(q$)goto2040,2190,2250,23
30,2430,2580,2700,2860,300
2030 goto2010
2040 rem ****************************
2050 rem ********* neueingabe t/z *
2060 rem ****************************
2070 pn=len("Neueingabe eines Files
")
2080 scnclr:char1,(40-pn)/2,2,"Neue
ingabe eines Files",1:print:gosub66
0
2090 gosub680:gosub660
2100 poke19,4:input"File Name: ";fl
$:poke19,0:print:ifn=1000thenn=0
2110 print"EG-Text...";:inputsb$(n)
:print
2120 ifsb$(n)=chr$(42)then1890
2130 print"EG-Zahl...";:inputz(n):p
rint
2140 n=n+1
2150 goto2110
2160 rem ****************************
2170 rem ************* file laden *
2180 rem ****************************
2190 scnclr:printchr$(18)"File lade
n."chr$(146)
2200 poke19,4:input"File-Name: ";fl
$:poke19,0:print
2210 open2,8,2,fl$+",s,r":input#2,n
2220 fori=0ton:input#2,sb$(i):input
#2,z(i):next:close2
2230 goto1890
2240 rem ****************************
2250 rem ********* file speichern *
2260 rem ****************************
2270 scnclr:printfl$" wird gesicher
t. Bitte warten..."
2280 open2,8,2,"$0:"+fl$+",s,w"
2290 print#2,n
2300 i=0:dountili=n:print#2,sb$(i):
print#2,z(i):i=i+1:loop:close2
2310 goto1890
2320 rem ****************************
2330 rem ********** file anzeigen *
2340 rem ****************************
2350 scnclr:printchr$(18)"File anze
igen."chr$(146)
2360 print"File-Name: "fl$
2370 iffl$=""then2680
2380 i=0:dountili=n
2390 printsb$(i)tab(25);:print usin
g dz$;z(i):i=i+1:loop
2400 gosub2870
2410 goto1890
2420 rem ****************************
2430 rem *************** addition *
2440 rem ****************************
```

```
2450 scnclr:printchr$( 18)"Werte add
ieren."chr$( 146)
2460 print"File-Name: "fl$
2470 iffl$=""then2680
2480 s=0:i=0:dountili=n
2490 printsb$(i)tab(25);:print usin
g dz$;z(i):s=s+z(i):i=i+1:loop
2500 printtab(27)"--------":printta
b(25);:print using dz$;s
2510 print:print:printtab(10)chr$( 1
8)"Ausdrucken j/n ?"chr$( 146)
2520 getx$:ifx$="n"then1890
2530 ifx$="j"then2700
2540 goto2520
2550 gosub2870
2560 goto1890
2570 rem ****************************
2580 rem ***************** append *
2590 rem ****************************
2600 scnclr:printchr$( 18)"File erwe
itern/Daten anhaengen"chr$( 146)
2610 print"File-Name: "fl$
2620 iffl$=""then2680
2630 printsb$(n-1),z(n-1)
2640 :input"EG-Text...:";sb$(n):pri
nt:ifsb$(n)=chr$( 42)then1890
2650 input"EG-Zahl...:";z(n):print
2660 n=n+1
2670 goto2640
2680 char1,7,12,"Keine Daten im Spe
icher!",1:forq=1to300:nextq:goto189
0
2690 rem ****************************
2700 rem ********* druckerroutine *
2710 rem ****************************
2720 scnclr:printchr$( 18)"Daten dru
cken."chr$( 146)
2730 print"Aenderung der"
2740 print"Sekundaeradresse:"
2750 printtab( 19)chr$( 145)"7"chr$( 1
57)chr$( 157)chr$( 157);
2760 inputsa
2770 print:printchr$( 18)"Wenn Druck
er bereit ist, Taste druecken."chr$
( 146)
2780 poke239,0:wait239,1
2790 open4,4,sa:cmd4
2800 print#4,tab(5)fl$
2810 fori=0ton-1
2820 print#4,tab(5)sb$(i)tab(40-len
(sb$(i))-len(dz$));:print#4,using d
z$;z(i):s=s+z(i):next
2830 print#4,tab(37)"--------"
2840 print#4,tab(45-len(dz$));:prin
t#4,using dz$;s
2850 print#4:close4:goto1890
2860 scnclr:printchr$( 12)chr$( 142):
end
2870 printchr$( 17)chr$( 17)chr$( 18)"
Taste fuer 'MENUE' "chr$( 146)
2880 poke239,0:wait239,1
2890 return
2900 rem mixed datei ===========c16
2910 rem 060671 bytes memory     ===
2920 rem 008731 bytes program    ===
2930 rem 000000 bytes variables  ===
2940 rem 000000 bytes arrays     ===
2950 rem 000000 bytes strings    ===
2960 rem 051625 bytes free (0)   ===
2970 rem ===========================
```

# TIPS & TRICKS
## CURSOR MIT PRINT & SYS

Vergessen Sie die umständliche Positionierung des Cursors mit den Steuerzeichen. Es gibt einfachere und leistungsfähigere Möglichkeiten, dieses zu realisieren. Außerdem ist die Möglichkeit mit den Steuerzeichen langsam und frißt Speicherplatz. Das Basic 3.5 des C16/C116/P4 stellt uns hierzu den Befehl CHAR zur Verfügung. Obwohl er laut Handbuch im Grafikbereich angesiedelt ist, läßt er sich doch in jedem Programm verwenden. Allerdings gibt es, aus eigener Erfahrung, gelegentlich Schwierigkeiten mit diesem Befehl. Für eine andere Möglichkeit stellt uns das Betriebssystem noch eine andere Routine zur Verfügung. Sie nennt sich Plot. Der Computer braucht sie laufend für die gleichen Aufgaben. Warum sollen wir sie dann nicht benutzen? Obwohl man bei der 16er-Serie ohne Poke-Befehle auskommen kann, müssen wir ihn hier anwenden. Auch ein SYS-Befehl wird benötigt (Sys steht immer für System). Aber die Sache ist halb so schlimm wie es aussieht. Die Einsprungadresse beträgt auch hier 65520. Wieso auch hier, werden Sie sich fragen. Da Commodore auf Tradition hält, nicht lachen, sind die Einsprungadressen des Kernal immer gleich. Für die Routine mit SYS benötigen wir 3 Adressen:

2035: für die Zeile
2036: für die Spalte
65520: Einsprungadresse der Routine

Geben Sie einmal folgendes Programm ein und sehen Sie, was sich hier tut.

```
100 CU=65520
110 ZE=2035
120 SP=2036
130 PokeZE,11:PokeSP,
    10:SYS CU
140 Print 'C B M Revue'
```

Wie Sie sehen, wird jeder Printbefehl der Anweisung an die mit ZE und SP gewählte Stelle des Bildschirms gesetzt. Achtung! Denken Sie immer daran, daß der Computer mit der Zählerei bei 0 beginnt!

# KALENDER

Das Programm „Kalender“ ist ein Utilitie, welches sehr praktisch ist, und dem Computerfan erlaubt, seine Jahresplanung ganz auf das Terminal zu verlegen. Ob man nun nur die Monate auf Feiertage durchsehen will, oder nachprüft, auf welchen Wochentag diesmal Weihnachten fällt, alles ist selbstverständlich der Grundstock für dieses Programm.

## VIELES IST MÖGLICH!

Interessant für den User sind die weiteren Möglichkeiten, die ihm „Kalender“ bietet: Er kann sowohl direkt in den Monat, als auch, was schneller und praktischer geht, im Direktmodus Notizen schreiben. Diese Notizen tauchen dann selbstverständlich beim Auflisten des jeweiligen Monatsquartals oder in der Gesamtübersicht, mit entsprechendem Datum, auf. Hat man nun alle seine Notizen eingegeben, kann man diese separat auf Band speichern, und auch später wieder einlesen lassen. Die Notizen müssen nicht alle in einem Block eingegeben werden, es geht auch nacheinander, nur müssen sie dann auch nacheinander in Blöcken gelesen werden. Als besonderen Service gibt dieses Programm auch eine Erklärung seiner Besonderheiten und eine Hilfe bei auftretenden Schwierigkeiten.
Insgesamt gesehen ist dieses Programm „Kalender“ bei richtiger Anwendung sehr nützlich für den User. Es würde bestimmt noch effektiver werden, wenn man statt der verwendeten Datasettenversion die Diskettenversion nähme, doch da ich Besitzer einer Datasette bin, habe ich mich für erstere Lösung entschieden.

## Zur Programmerklärung:

Zeile 270:
Um dem User Arbeit zu ersparen, sind die programmierbaren Tasten F1, F2, F3, F8 (“Help“) mit viel gebräuchlichen „Notizfloskeln“ belegt.
Zeile 390:
Falls irgendwo ein Fehler auftreten sollte (z.B. Run Stop oder “Type Mismatch“) springt das Programm zum Menü zurück.
Zeile 420:
Durch die Zahl, die gewählt wurde, springt der Computer in das entsprechende Unterprogramm.
Zeile 500–610:
Bei der Monatseingabe kann man entweder den ganzen Monatsnamen oder dessen Ordnungszahl nennen: daß der Monatsstring MO$ im Endeffekt immer den ganzen Monat beinhalten muß, hängt mit dem späteren Ausdruck zusammen.
Zeile 630:
Hier wird dem Computer durch die Variable GS mitgeteilt, ob das Unterprogramm direkt vom Unterprogramm “Notiz – in Monat“ oder vom Menü angesprungen wurde, und handelt auch dementsprechend.
Zeile 670–690:
Hier wird die Variable DA so zurechtgebogen, daß man die Notiz grafisch auch wirklich an dem Tag eingeben kann, den man vorher angegeben hat.
Zeile 870/990/1110:
Diese Zeilen bewirken, daß der Monatsname, unabhängig von seiner Länge, immer in der Mitte der Zeile steht.
Zeile 880 f.f./1000 f.f./1120 f.f.:
Mit diesen Zeilen wird der Monatsname, wieder unabhängig von seiner Länge, unterstrichen.
Zeile 1250–1400:
Jede Datazeile beinhaltet die Sonn- und Feiertage eines bestimmten Monats.

*Bitte lesen Sie weiter auf Seite 113*

```
10 rem Kalender 1986 ============P4
20 rem (P)          CBM REVUE TEAM  =
30 rem ==============================
40 rem (C)        by               =
50 rem Martin Purucker             =
60 rem Wuerzburg                   =
70 rem Version 3.5     40Z/ASCII   =
80 rem Plus/4 C-16/116 + 1531      =
90 rem ==============================
250 dimte$(31,12)
260 ro$=chr$(18):ra$=chr$(146):es$=
chr$(27):q$=chr$(81)
270 key 1,"geburtstag":key 2,"termi
n":key 3,"besprechung":key8,"party"
280 rem Maske *******************
290 scnclr:color 1,7,4:color 4,12,6
:color 0,12,6:wo$="":gs=0
300 char 1,5,1,"kalender 1986 - ueb
ersicht :"
310 char 1,5,5,ro$+" 1 "+ra$+".....
monate/quartal"
320 char 1,5,7,ro$+" 2 "+ra$+".....
notiz - direkt"
330 char 1,5,9,ro$+" 3 "+ra$+".....
notiz - in monat"
340 char 1,5,11,ro$+" 4 "+ra$+"....
.notiz - speichern"
350 char 1,5,13,ro$+" 5 "+ra$+"....
.notiz - lesen (von band)
360 char 1,5,15,ro$+" 6 "+ra$+"....
.notiz - gesamtuebersicht"
370 char 1,5,17,ro$+" 7 "+ra$+"....
.programm - ende"
380 char 1,5,19,ro$+" 8 "+ra$+"....
.programm - erklaerungen"
390 trap 290:char 1,28,24,"(c) m.p.
85"
400 poke 239,0:getkey no:color 1,1:
color 0,2:color 4,7,6
410 if no>8orno<1then290
420 on no gosub450,720,650,1410,146
0,1790,440,1850
430 goto 290
440 scnclr:end:rem Programmende ***
450 rem Monats/Quartal Eingabe ****
460 scnclr
470 print:input"gewuenschter monat.
";mo$
480 print:input"gewuenschtes quarta
l.";qu
490 if qu>3 or qu<1 then 480
500 ifmo$="januar"or mo$="1"thenmo=
1:mo$="januar"
510 ifmo$="februar"or mo$="2"thenmo
=2:mo$="februar"
520 ifmo$="maerz"or mo$="3"thenmo=3
:mo$="maerz"
530 ifmo$="april"or mo$="4"thenmo=4
```

```
:mo$="april"
540 ifmo$="mai"or mo$="5"thenmo=5:m
o$="mai"
550 ifmo$="juni"or mo$="6"thenmo=6:
mo$="juni"
560 ifmo$="juli"or mo$="7"thenmo=7:
mo$="juli"
570 ifmo$="august"or mo$="8"thenmo=
8:mo$="august"
580 ifmo$="september"or mo$="9"then
mo=9:mo$="september"
590 ifmo$="oktober"or mo$="10"thenm
o=10:mo$="oktober"
600 ifmo$="november"or mo$="11"then
mo=11:mo$="november"
610 ifmo$="dezember"or mo$="12"then
mo=12:mo$="dezember"
620 on qu gosub 850,970,1090
630 if gs=1 then return
640 poke 239,0:getkey a$:return
650 rem Notiz - in Monat **********
660 gs=1:gosub 450:color 1,1:char 1
,29,1,"":input da:de=da
670 if qu=1thenda=da*2+2
680 if qu=2thenda=(da-10)*2+2
690 if qu=3thenda=(da-20)*2+2
700 char 1,5,da,es$+q$:inputte$(de,
mo)
710 getkey a$:return
720 rem Notiz - Direkt ************
730 scnclr
740 print:print"in welchem monat so
ll die notiz stehen":print
750 print"(1-12) ";:input mo
760 if mo>12 or mo<1 then 750
770 print:print"an welchem tag soll
 die notiz stehen":print:print"(1-3
1/30/28) ";
780 input de
790 if de>31 or de<1 then print"(1-
31/30/28) ";:goto 780
800 print:print"wie soll die notiz
lauten (-32 zeichen)":print:inputte
$(de,mo)
810 iflen(te$(de,mo))>32 then print
"(-32 zeichen) ":inputte$(de,mo)
820 print:print"noch weitere notize
n (j/n) ?":getkey nn$
830 if nn$="j"then720
840 return
850 rem monate 1.quartal **********
860 scnclr
870 color 1,6,4:char 1,20-int(len(m
o$)*.5),1,mo$
880 fort=1tolen(mo$)
890 char 1,20-int(len(mo$)*.5)-1+t,
2,"D"
900 next
910 char 1,0,1,str$(quartal)+". qua
rtal"
920 fort=2to11:color 1,6,4:gosub 12
50:gosub 1520
930 char1,0,t*2,wo$+str$(t-1):for s
=5to38:char1,s,t*2,ro$+" "+ra$
940 ifte$(t-1,mo)<>""then char 1,6,
t*2,te$(t-1,mo)
950 next:next
960 return
970 rem Monate 2.Quartal **********
980 scnclr
990 color 1,6,4:char 1,20-int(len(m
o$)*.5),1,mo$
1000 fort=1tolen(mo$)
1010 char 1,20-int(len(mo$)*.5)-1+t
,2,"D"
1020 next
1030 char 1,0,1,str$(quartal)+". qu
artal"
1040 fort=12to21:color 1,6,4:gosub
1250:gosub 1520
1050 char1,0,(t-10)*2,wo$+str$(t-1)
:for s=5to38:char1,s,(t-10)*2,ro$+"
."+ra$
1060 ifte$(t-1,mo)<>""then char 1,6
,(t-10)*2,te$(t-1,mo)
1070 next:next
1080 return
1090 rem Monate 3.Quartal *********
1100 scnclr
1110 color 1,6,4:char 1,20-int(len(
mo$)*.5),1,mo$
1120 fort=1tolen(mo$)
1130 char 1,20-int(len(mo$)*.5)-1+t
,2,"D"
1140 next
1150 ifmo$="januar"ormo$="maerz"orm
o$="mai"ormo$="juli"ormo$="august"t
hentage=32
1160 ifmo$="oktober"ormo$="dezember
"thentage=32
1170 ifmo$="april"ormo$="juni"ormo$
="september"ormo$="november"thentag
e=31
1180 ifmo$="februar"thentage=29
1190 char 1,0,1,str$(quartal)+". qu
artal"
1200 fort=22totage:color 1,6,4:gosu
b1250:gosub1520
1210 char1,0,(t-20)*2,wo$+str$(t-1)
:for s=5to38:char1,s,(t-20)*2,ro$+"
."+ra$
1220 ifte$(t-1,mo)<>"" then char 1,
6,(t-20)*2,te$(t-1,mo)
1230 next:next
1240 return
1250 rem feiertagsentscheidung ****
```

```
126Ø restore128Ø+1Ø*mo
127Ø for vv=1to8:readx:ifx=t-1thenc
olor1,3,4:return
128Ø next:return
129Ø data 1,5,6,12,19,26,Ø,Ø
13ØØ data 2,9,16,23,Ø,Ø,Ø,Ø
131Ø data 2,9,16,23,28,3Ø,31,Ø
132Ø data 6,13,2Ø,27,Ø,Ø,Ø,Ø
133Ø data 1,4,8,11,18,19,25,29
134Ø data 1,8,15,17,22,29,Ø,Ø
135Ø data 6,13,2Ø,27,Ø,Ø,Ø,Ø
136Ø data 3,8,1Ø,15,17,24,31,Ø
137Ø data 7,14,21,28,Ø,Ø,Ø,Ø
138Ø data 5,12,19,26,Ø,Ø,Ø,Ø
139Ø data 1,2,9,16,19,23,3Ø,Ø
14ØØ data 7,14,21,25,26,28,Ø,Ø
141Ø rem Notiz - Speichern ********
142Ø scnclr:open 1,1,1
143Ø for c=1to31:forcc=1to12
144Ø if te$(c,cc)<>""then print#1,c
;",";cc;",";te$(c,cc)
145Ø next:next:print#1,Ø;",";Ø;",";
a9$:close1:return
146Ø rem Notiz - Lesen ************
147Ø scnclr:open 1,1,Ø
148Ø do
149Ø input#1,c,cc,te$(c,cc)
15ØØ loopuntilc=Øandcc=Ø
151Ø close 1:return
152Ø rem wochentagsentscheidung ***
153Ø ifmo=4andqu=3ormo=7andqu=3ormo
=8andqu=2ormo=9andqu=1ormo=12andqu=
1thenwe=1
154Ø ifmo=1andqu=3ormo=2andqu=2ormo
=3andqu=2ormo=4andqu=1ormo=7andqu=1
thenwe=2
155Ø ifmo=1Øandqu=3ormo=11andqu=2th
enwe=2
156Ø ifmo=1andqu=1ormo=5andqu=3ormo
=6andqu=2ormo=1Øandqu=1thenwe=3
157Ø ifmo=5andqu=1ormo=8andqu=3ormo
=9andqu=2ormo=12andqu=2thenwe=4
158Ø ifmo=2andqu=3ormo=3andqu=3ormo
=4andqu=2ormo=7andqu=2ormo=8andqu=1
thenwe=5
159Ø ifmo=11andqu=3thenwe=5
16ØØ ifmo=1andqu=2ormo=2andqu=1ormo
=3andqu=1ormo=6andqu=3ormo=1Øandqu=
2thenwe=6
161Ø ifmo=11andqu=1thenwe=6
162Ø ifmo=5andqu=2ormo=6andqu=1ormo
=9andqu=3ormo=12andqu=3thenwe=7
163Ø zn=171Ø+we*1Ø
164Ø ifwo$="so"then zn=172Ø
165Ø ifwo$="mo"then zn=173Ø
166Ø ifwo$="di"then zn=174Ø
167Ø ifwo$="mi"then zn=175Ø
168Ø ifwo$="do"then zn=176Ø
169Ø ifwo$="fr"then zn=177Ø
17ØØ ifwo$="sa"then zn=178Ø
171Ø restore zn:read wo$:return
172Ø data "mo"
173Ø data "di"
174Ø data "mi"
175Ø data "do"
176Ø data "fr"
177Ø data "sa"
178Ø data "so"
179Ø rem Notiz-Gesamtuebersicht ***
18ØØ scnclr:char 1,5,1,"notiz gesam
tuebersicht":print:print
181Ø forcc=1to12:forc=1to31
182Ø ifzv=1Øthenchar1,34,24,ro$+"sp
ace"+ra$:poke239,Ø:getkeya$:zv=Ø:sc
nclr:print
183Ø if te$(c,cc)<>""thenprintc;"."
;cc;". : ";te$(c,cc):zv=zv+1
184Ø next:next:char 1,34,24,ro$+"sp
ace"+ra$:poke239,Ø:getkey a$:return
185Ø rem Programm - Erklaerungen **
186Ø scnclr
187Ø char 1,8,1,"programmerklaerung
en :"
188Ø char 1,1,3,"- quartal : die mo
nate sind in 3 quar-"
189Ø char 1,13,5,"tale eingeteilt:"
19ØØ char 1,13,7,"1.quartal=tage 1-
1Ø"
191Ø char 1,13,9,"2.quartal=tage 11
-2Ø"
192Ø char 1,13,11,"3.quartal=tage 2
1-31/3Ø/28."
193Ø char 1,1,13,"- ":color 1,3,4:c
har 1,3,13,ro$+"          "+ra$
194Ø color 1,1:char 1,11,13,": rot
gezeichnet werden"
195Ø char 1,13,15,"alle feier- und
sonntage,"
196Ø char 1,13,17,"sowie notizen, d
ie diese"
197Ø char 1,13,19,"tage betreffen."
198Ø char 1,34,24,ro$+"space"+ra$
199Ø poke 239,Ø:getkey a$:scnclr
2ØØØ char 1,1,2,"- fehler ":char 1,
1Ø,2,": falls im programm irgend-"
2Ø1Ø char 1,12,4,"ein unverstaendli
cher fehler"
2Ø2Ø char 1,12,6,"auftreten sollte,
 zeile 37Ø"
2Ø3Ø char 1,12,8,"loeschen, denn do
rt werden"
2Ø4Ø char 1,12,1Ø,"auch syntax erro
rs mit einem"
2Ø5Ø char 1,12,12,"ruecksprung ins
menue unter-"
2Ø6Ø char 1,12,14,"drueckt, und das
```

```
 programm"
2070 char 1,12,16,"normal durchlauf
en lassen -"
2080 char 1,12,18,"sofort wird der
fehler auf-"
2090 char 1,12,20,"gedeckt."
2100 char 1,34,24,ro$+"space"+ra$
2110 poke 239,0:getkey a$:scnclr
2120 char 1,1,2,"- eingabe : nicht
vergessen ! -"
2130 char 1,13,4,"'input' nimmt all
es bis"
2140 char 1,13,6,"auf komma und dop
pelpunkt."
2150 char 1,34,24,ro$+"space"+ra$
2160 poke 239,0:getkey a$:return
2170 rem Kalender 1986 =========3.5
2180 rem 060671 Bytes Memory    ===
2190 rem 000000 Bytes Program   ===
2200 rem 000000 Bytes Variables ===
2210 rem 000000 Bytes Arrays    ===
2220 rem 000000 Bytes Strings   ===
2230 rem 000000 Bytes Free (1)  ===
2240 rem ==========================
```

*Fortsetzung von Seite 110*

# KALENDER

Durch "Restore X" wird der Lesezeiger auf den jeweils betrachteten Monat gestellt.
Zeile 1450:
Durch den seltsam anmutenden Befehl "Print#1,0; ","; 0; ","; A9$ wird dem Computer dann beim Dateneinlesen "gesagt", daß er aus der Do-Loop-Schleife herausgehen soll.
Zeile 1520–1780:
Die IF-Abfragen finden heraus, mit welchem Wochentag der jeweilige Monat beginnt, und mit den Zeilen 1630–1780 wird dann einfach durchgezählt.
Zeile 1820:
Wenn in der Notizgesamtübersicht 10 Notizen ausgedruckt sind, wartet das Programm auf einen Tastendruck zur Fortsetzung.
Zeile 1850–ENDE:
Dieses Programmteil ist nur für die allgemeine Erklärung des Programms und zur Lösung von Schwierigkeiten da. Hat man nämlich einen Fehler gemacht, klärt einen der Punkt "Fehler" auf, daß dieser durch "Trap" nicht zum Vorschein kommt, sondern das Programm einfach zum Menü zurückspringt.
Falls Sie diesen Programmteil trotzdem für überflüssig halten, müssen Sie folgendes tun:
1. Zeile 380 löschen
2. In Zeile 410 statt "IF NO>8" "IF NO>7" ausbessern.
3. In Zeile 420 die letzte anzuspringende Zeilennummer und das Komma davor löschen.
4. Zeile 1850 bis 2160 löschen.
Dann funktioniert alles ohne Programmabschnitt "Erklärungen".
Stringliste:
RO$ = CHR$(18)= Revers On
RA$ = CHR$(146)= Revers Off
ES$ = CHR$(27)=Escape
Q = CHR(81)=Q

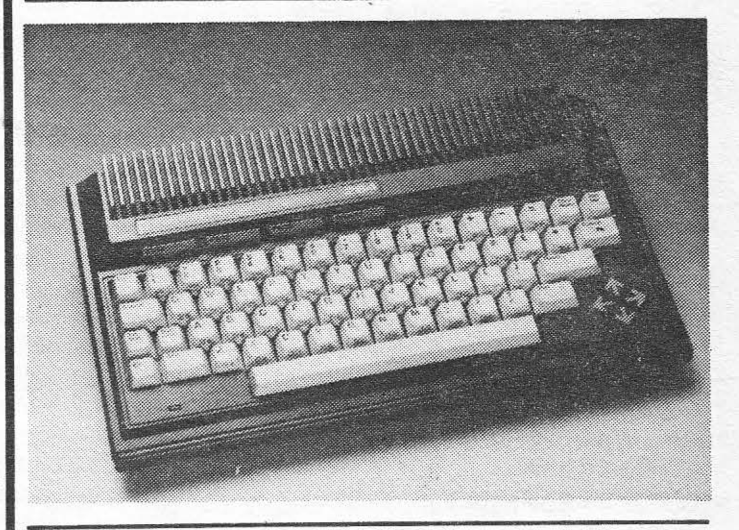
*

Anmerkung:
ESC+Q$ löschen die Zeile ab dem Cursor.

*Martin Purucker*

# TIPS & TRICKS P/4

Der 'Plus 4' ist ein bewundernswertes und begeisterndes Gerät und für alle ernsthaften Anwender das Beste, was zu empfehlen ist (Gegensatz: Seitenlange Programme zu Hause auswendig einpauken,um im Warenhaus den Zuschauern etliche bunte Kreise auf den Bildschirm zu machen – gibt es etwas Dümmeres? Kleine Männchen umherhüpfen zu lassen – ist das im Sinne der Erfinder?).



Gerade die Interessengruppen, für die der Hersteller dieses Gerät gemacht hat, wissen leider nichts von ihm, da sie – zwar interessiert, doch zurückhaltend – in dem noch herrschenden Dschungeldunkel nicht klar sehen und eine Werbung des Herstellers fast völlig fehlt.
Eines der im Plus 4 integrierten und im Anleitungsbuch bestens erklärten Programme ist die sog. Dateiverwaltung. Hierin gibt es den 'RV'-Befehl, um die Dateien schnell durchlaufen zu lassen. 'Schnell' ist hier fast zu schnell, während der zugehörige Stoppbefehl recht langweilig reagiert. Es gibt eine einfache Abhilfe mit dem 'Search'-Befehl, der üblicherweise mit der Eingabe eines Suchbegriffes verbunden ist. In diesem Falle wird das Eingeben eines Suchbegriffes unterlassen und statt dessen ein zweites 'Return" gedrückt. Das bewirkt, daß Nr. 1 der unsortierten oder einer sortierten Datei auf dem Bildschirm erscheint. Gleichzeitig fragt der C.: 'Yes/No?' – bei 'Yes' kommt Nr. 2 der Datei. So kann die ganze Datei-Diskette abgefragt werden – so schnell oder so langsam, wie jeweils 'Yes' gedrückt wird.
Das Anleitungsbuch gibt weiterhin keine Auskunft über den Fall, daß ein einziger Brief, d.h. eine einzige Adresse aus der Dateiverwaltung, verwendet bzw. gedruckt werden soll: Es wird vor der Angabe 'tf;:rc' die RC-Nr. der zu verwendenden Adresse eingetippt, woraufhin der Druckvorgang bei Textende endet, da ja kein 'eof' eingegeben wurde. Andernfalls muß rechtzeitig 'Run/Stop' betätigt werden.
Schließlich ist noch zu erwähnen, daß der Cursor mit 'Shift' und '=' in Spalte 77 gebracht werden kann.
Ein Computer für einen rationalen Zweck ist wesentlich unterhaltsamer als einer nur zur Unterhaltung!

*Helmut Kopplin*

# SO GRÜNDEN SIE EINEN COMPUTER-CLUB

**Mit wachsender Verbreitung der Computer steigt auch die Zahl der Anwender, die ihr Hobby nicht nur im dunklen Kämmerlein betreiben wollen, sondern sich mit Gleichgesinnten treffen und Erfahrungen, Meinungen und Tricks austauschen wollen. Daß dabei fast immer eine Clubgründung erfolgt, liegt wohl in der Natur der Deutschen (ein geflügelter Satz lautet nicht umsonst: Wo zwei Deutsche mit gleichen Interessen zusammentreffen, gründen sie einen Verein ...).**

Die Redaktion erhält immer mehr Zuschriften von Usern, die Tips und Anregungen zu einer Clubgründung haben wollen. Stellvertretend für alle haben wir hier den Brief eines Lesers abgedruckt. „Wir haben hier, meines Wissens, keinen Verein oder Club, der sich mit Computern befaßt. Das möchte ich ändern.

Es stellen sich jedoch einige Fragen, die ich unmöglich alleine beantworten kann. Ich bitte deshalb um Ihre Hilfe.

1. Was muß ich berücksichtigen?
2. Was darf ich tun, ohne mit dem Gesetz in Konflikt zu geraten?
3. Woher bekomme ich Informationen?
4. Wo finde ich Ansprechpartner?
5. Was hilft mir bei Startschwierigkeiten?“



**ZU FRAGE 1:**

Generell muß man sagen, daß es kein Patentrezept für eine erfolgreiche Clubgründung gibt. Man kann unmöglich davon ausgehen, daß zehn Leute mit dem gleichen Hobby auch den gleichen Charakter haben. Der Mensch ist immer noch der größte Unsicherheitsfaktor. Berücksichtigen müssen Sie vor allem, daß Sie sich nicht mit jedem Computer beschäftigen können. Sie sollten sich schon auf einen oder zwei Rechner spezialisieren. Sagen wir, als Hauptobjekt ein Commodore 16 und nebenbei noch den Atari 520 ST oder ein MSX-Modell. Zu viele Computer lassen ein effektives Arbeiten nicht zu. Natürlich soll das auf keinen Fall heißen, daß Besitzer anderer Computer nicht Mitglied werden dürfen.

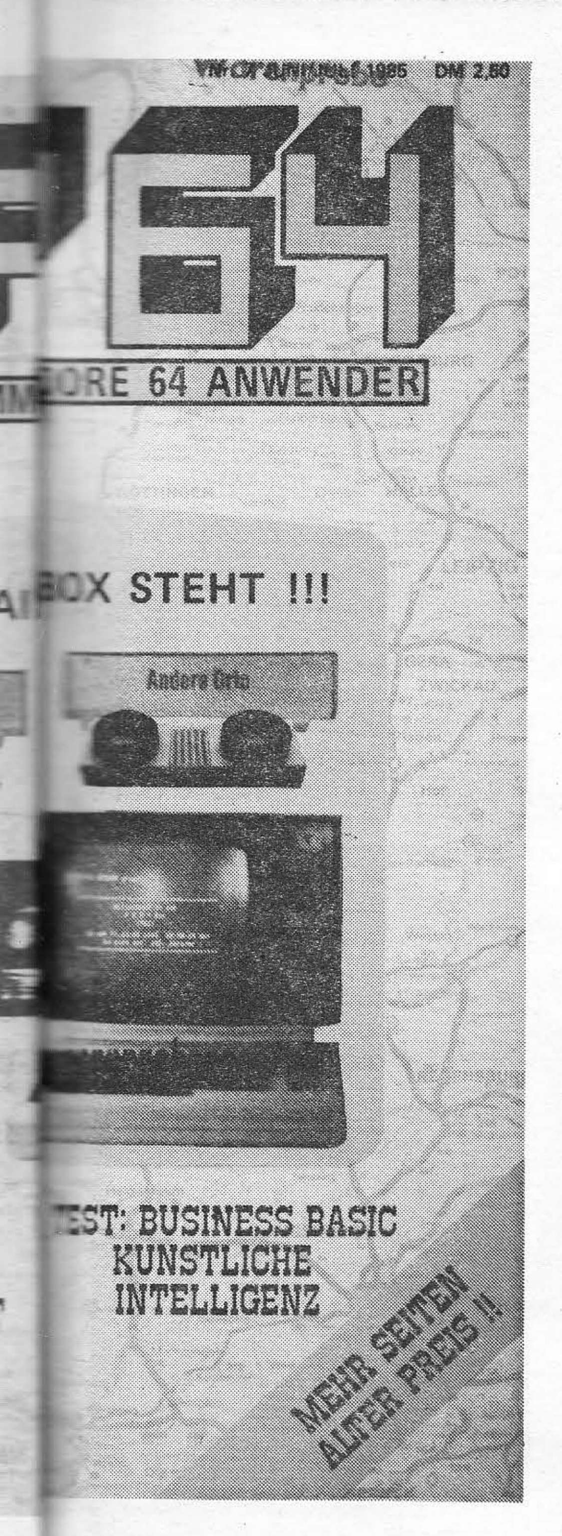
Weiterhin sollten Sie eine Eintragung ins Vereinsregister vornehmen und die Gemeinnützigkeit beantragen.

Letzteres hat den Vorteil, daß Sie von der Stadt Räumlichkeiten und Geldmittel (soweit vorhanden!!!) erhalten können, die Ihnen als „lose“ Vereinigung nicht zustehen.

Aber Vorsicht: Sie dürfen keinen Firmennamen im Titel führen. Dieses Problem machte z.B. dem C64-Anwender-Club München zu schaffen. Eine Gemeinnützigkeit wurde mit der Begründung verweigert, im Titel würde Werbung für einen Computer gemacht. Der Club zieht jetzt die Konsequenzen und wird sich erstens demnächst umbenennen und zweitens für andere Computer offener werden.




**Wenn die „richtige Crew“ dahintersteht, kann ein Computer-Club sehr aktiv werden. Hier etwa die Clubzeitschrift des 64er-Clubs München, die jedes der Mitglieder selbstverständlich kostenlos erhält. Darüber hinaus aber wird diese Clubzeitung sogar auf Münchener Bahnhöfen und in bestimmten Computershops verkauft! Mit den Erlösen aus dem Verkauf und den Anzeigen finanziert der Club diese Zeitschrift.**




## ZU FRAGE 2:

Zäumen wir das Pferd einmal von hinten auf: Was dürfen Sie nicht tun? Am wichtigsten und deshalb zuerst: keine geschützten Programme kopieren, Kopien verkaufen, tauschen und in einer Programmbibliothek führen. Letzteres gilt nicht für Programme, die von Softwarefirmen offiziell und zur Information dem Club geschenkt worden sind.

Führen Sie Sammelbestellungen durch, müssen Sie mit dem Umsatz unter einer bestimmten Grenze bleiben, die sie am besten vom örtlichen Finanzamt erfragen. Übersteigen Ihre Geschäfte diesen Betrag, werden Sie gegenüber dem Finanzamt anzeigepflichtig, müssen Buch führen und sogar Umsatzsteuer zahlen.

Das sind die wichtigsten Einschränkungen, die Sie beachten müssen. Nicht verboten ist, eigene Software zu entwickeln und zu verkaufen, Schulungen durchzuführen, Clubtreffen zu veranstalten etc. Sie werden sehen, daß zwanzig Leute auch zwanzig unterschiedliche Ideen haben, die Sie verwerten können.

## ZU FRAGE 3:

Sie sprechen da ein heikles Thema an, geben doch manche Hersteller gerade soviel Auskunft wie eine Betonmauer, nämlich keine. Dabei hat jede Firma ihre(n) Clubbeauftragte(n), die (der) gerade dann auf Geschäftsreise ist, wenn man sie (ihn) dringendst braucht. Allerdings ändert sich diese Situation langsam, und dazu haben die Clubs einen guten Teil beigetragen.

Wenn Sie also Informationen haben wollen, wenden Sie sich an die entsprechende Person und bleiben Sie um Gottes Willen hartnäckig. Übrigens haben die meisten Hersteller eine sog. Hot-Line, unter der immer jemand zu erreichen ist.

Informationen allgemeiner Art erhalten Sie natürlich aus den diversen Fachzeitschriften. Natürlich können Sie nicht alle abonnieren, die für „Ihre" Computer wichtigsten sollten Sie aber führen. Weiterhin ist es von Vorteil, sich einen Fachhändler zu suchen, über den Sie alle Bestellungen zu günstigen Konditionen (sprich Rabatte, Freiexemplare) abwickeln können. Dieser kann Ihnen sicher das ein oder andere Datenblatt und Schaltpläne besorgen.

## ZU FRAGE 4:

Wie oben schon erwähnt, finden Sie Ansprechpartner bei den Herstellern. Sehr unüblich, aber sehr effizient, ist der Weg über einen Schwesterclub. Sie suchen sich einen Club mit ähnlichen Interessen und tauschen Informationen, Ideen und Anregungen aus. Dazu ist ein Treffen nicht zwingend notwendig, das Telefon oder die Clubzeitung tun's auch.

## ZU FRAGE 5:

Probleme und Startschwierigkeiten wird es immer geben, auch hier gibt es keine Regeln, nach denen man vorgehen muß. Deshalb ein paar wichtige Tips:

Hüten Sie sich vor einem: Seien Sie kein Hardware-Bastler, Starprogrammierer, Finanzgenie und Clubvorsitzender in Personalunion. Alles selbst machen zu wollen, geht auf jeden Fall schief. Splitten Sie den Club in sinnvolle Teilbereiche auf, etwa in die Bereiche Hardware, Software, Clubzeitung und Öffentlichkeitsarbeit. Auf der ersten Clubsitzung werden dann geeignete Leute für diese Bereiche bestimmt oder gewählt. Sie als Vorsitzender kontrollieren dann nur noch die Teilbereiche.

Meinen Sie nicht, Sie hätten den Erfolg gepachtet, mit Tiefschlägen müssen Sie rechnen.

Und ganz wichtig: Seien Sie mit ganzem Herzen bei der Sache. Entweder Sie führen konsequent den Club oder arbeiten dementsprechend dafür, oder Ihr „Clubschiff" wird unweigerlich an der nächsten Klippe zerschellen.

Wenn Sie das alles beherzigen, kann eigentlich nichts mehr schiefgehen, es sei denn ...

Es sei denn, Sie bleiben ein „Ein-Mann-Club", weil zu wenig Leute über Ihre Idee informiert sind. Deshalb noch ein kleiner Rat: Machen Sie auf sich aufmerksam.

Sie brauchen ja nicht gerade einen Zeppelin zu mieten, wie das einige Firmen tun (Fuji, Ashton Tate), ein kurzer Anruf bei Ihrer Zeitung kann da auch genügen. Oder Sie lassen Flugblätter drucken, die dann in Schulen (vorher Genehmigung beim Rektor holen!!!) und Computershops verteilt werden. Aber da fällt Ihnen sicher etwas ein.

Natürlich steht auch die COMMODORE-WELT mit Rat und Tat (soweit möglich) hilfreich zur Seite. Wenden Sie sich einfach an die Redaktion.

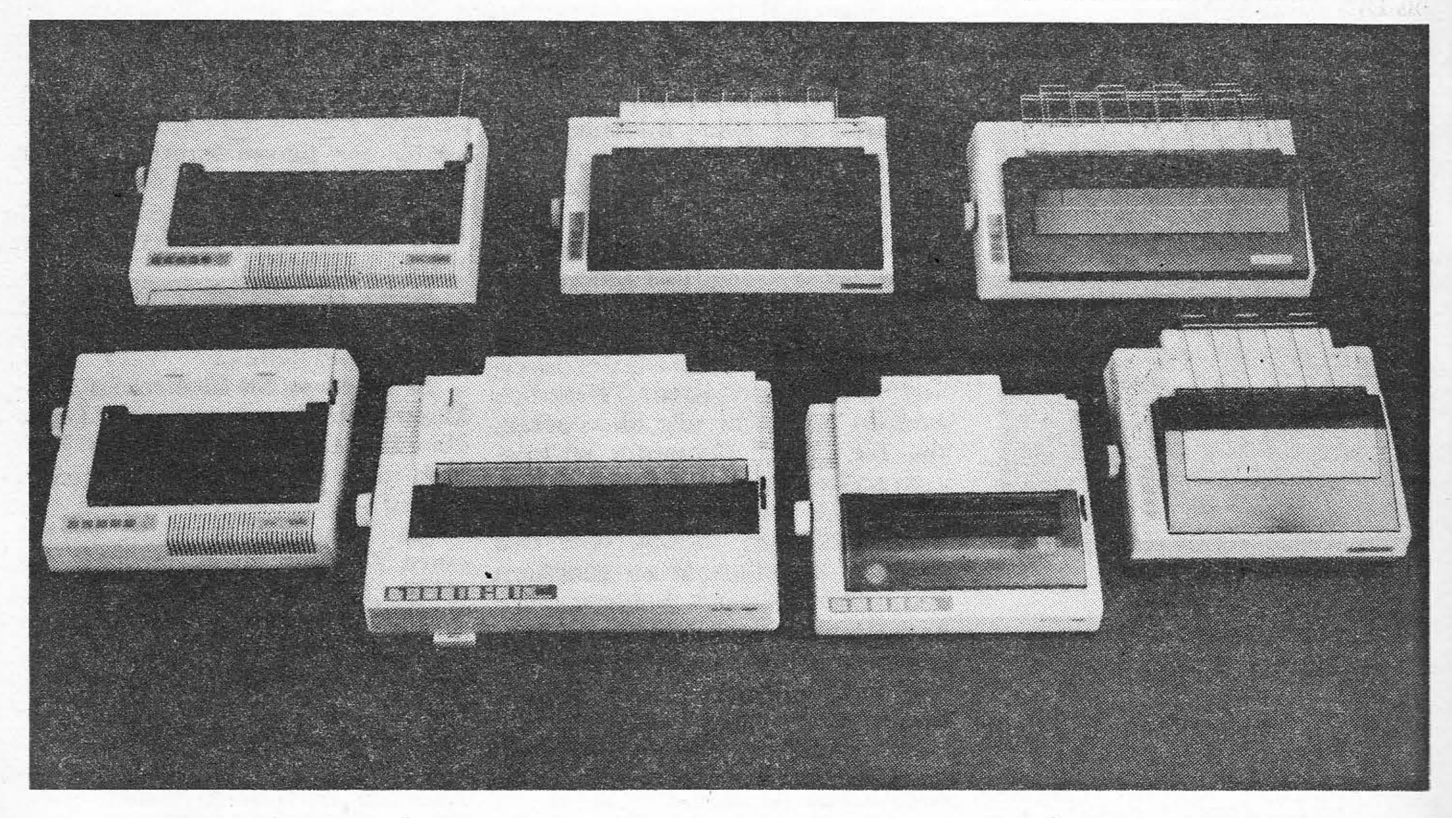
# SO FINDEN SIE DEN RICHTIGEN DRUCKER

**Neben der Floppy ist der Drucker das wichtigste Werkzeug für einen Computer. COMMODORE WELT sagt Ihnen, was Sie vor dem Kauf eines Druckers beachten sollten.**

Drucker gibt es mittlerweile schon wie Sand am Meer. Ob für den Heimgebrauch, zum Ausdruck für kleinere Programme und Listen oder aber für den Profi-Bereich: Für jede Nutzung gibt es mindestens ein halbes Dutzend Anbieter mit verschiedenen Leistungen und verschiedenen Preisen.

„Welche Marke und welchen Typ nehme ich, was ist für mich am Besten?" Diese oder ähnliche Fragen stellen sich jedem Computer-Benutzer, der sich einen Drucker kaufen will.

Selbstverständlich muß diese Frage im Endeffekt jeder einzelne für sich selbst entscheiden. Und gerade im Fall eines Druckerkaufes sollte man sich für diese Entscheidung besonders viel Zeit nehmen.
Das erste, was man tun sollte, ist, in den eigenen Geldbeutel zu schauen, um sich damit klarzuwerden, welche Summe überhaupt ausgegeben werden kann.
Jetzt erst sollte man sich informieren, welche Drucker für welchen Preis angeboten werden.
In der ersten Auswahl fallen alle Typen heraus, die mehr kosten, als Sie bereit oder fähig sind, dafür auszugeben.
In der zweiten Auswahl sollten Sie sich die Frage stellen, ob Ihr Drucker einen bestimmten Standard erfüllen soll.
Gängige Standards sind zum Beispiel Epson- und IBM-Druckformate. Die für den C16 geschriebene Druckersoftware ist entweder für einen Original Commodore Drucker (MPS-Serie) oder aber für einen Drucker mit Epson-Standard bestimmt; ebenso verhält es sich beim Apple IIe.
Haben Sie nun zum Beispiel eben einen C 16 und Sie sind sich sicher, daß Sie keinen IBM oder einen IBM-ähnlichen Rechner in nächster Zukunft kaufen wollen, so brauchen Sie einen Drucker mit Commodore- oder Epson-Standard.
Ansonsten sollte Ihr neuer Drucker sowohl den Epson- als auch den IBM-Standard beherrschen.
Ein weiteres Auswahlkriterium sollte die Grafikfähigkeit des Druckers sein. Ist er grafikfähig, so kann man mit diesem Gerät sowohl normalen Text als auch, mit einem Zusatzprogramm, hochaufgelöste Grafik-



**Es gibt sie wie Sand am Meer: Drucker. Hier allein das Angebot eines einzigen Unternehmens: Star.**

bilder ausdrucken. (Die meisten modernen Drucker sind grafikfähig.)
Ein weiterer, nicht minder wichtiger Punkt ist die Art der Schnittstelle, die der Printer besitzt.
Schnittstellen gibt es allgemein nur zwei verschiedene: Die serielle und die parallele Schnittstelle.
Die meisten Drucker im Home- und Personalcomputerbereich besitzen parallele Schnittstellen, eine Ausnahme bilden die Commodore.
Will man einen Drucker mit paralleler Schnittstelle an den C16/P4 anschließen, so benötigt man hierfür noch eine besondere Schnittstelle, ein Druckerinterface.
Bei manchen Angeboten ist diese Karte schon dabei, bitte beachten Sie dies!!! Ansonsten sind Mehrausgaben bis zu 250 und mehr Mark möglich!
Hat man diesen wichtigen Punkt auch beachtet, kann man endlich dazu übergehen, den Drucker nach seinen eigenen Leistungen zu testen.
Beginnen wir mit:

**a) Einzelblatteinzug:**
Einzelblatteinzug bedeutet nichts anderes, als daß Sie mit diesem Drucker ganz normales Papier (meist bis DIN A4-Format) benutzen können. Ein nicht zu unterschätzender Vorteil.

**b) Traktor bzw. Endlosblattverarbeitung:**
Dieser Zusatz bedeutet, daß der Drucker Endlospapier mit Lochrand verarbeiten kann.
Auch diese Eigenschaft ist von großer Bedeutung.
Selbstverständlich gibt es viele Drukker, die sowohl den Punkt (a) als auch den Punkt (b) erfüllen; sie sind den anderen Geräten vorzuziehen.

**c) Druckgeschwindigkeit:**
Die Druckgeschwindigkeit wird in Z/sec. = Zeichen pro Sekunde angegeben. Hat ein Drucker die Angabe 180 Z/sec. so bedeutet dies, daß man von einem Text mit 80 Zeichen pro Zeile jede Sekunde 180 Zeichen abdruckt.
Da die meisten Texte jedoch kürzere Zeilen haben, ist dieser Wert nicht hundertprozentig verbindlich, sondern zu hoch.
Dennoch ist ein Drucker mit einer Geschwindigkeit von 180 Z/sec. um einiges schneller als ein Drucker mit nur 120 Z/sec.

**d) NLQ=near letter quality= Schönschriftmodus:**
Ein durchschnittlicher, guter Drukker schreibt normalerweise mit einer 8x9 Punkte Matrix.
Schlechtere Drucker, wie zum Beispiel der Commodore MPS 801, haben nur eine 5x7 Punkte Darstellung.
Die Zeichen eines solchen Druckers sind weit schlechter zu lesen, also unsauberer, während die 8x9 Punkte-Drucker weitaus schönere Druckbilder erzeugen.
Achten Sie also vor dem Kauf auf die Druckmatrix Ihres Druckers!!!
Einige Drucker haben nun einen Extra-Modus, NLQ genannt.
Dieser Modus simuliert eine 18x24 Punkte Matrix; der Zeichenausdruck in diesem Modus ist, gegenüber dem Normalmodus, umwerfend; es ist kaum noch ein Unterschied zu einer Schreibmaschine feststellbar.
Dafür ist die Druckgeschwindigkeit auch erheblich verringert, was man aber für besonders wichtige Briefe in Kauf nehmen sollte.
Wichtig ist auf jeden Fall, daß der Drucker die Schönschreibmöglichkeit besitzt.

**e) Druckerpuffer:**
Ein Druckerpuffer ist ein Speicher, in dem Daten, die vom Computer kommen und die ausgedruckt werden sollen, zwischengespeichert = gepuffert werden.

Moderne Drukker ändern ihre Schriften heute per Einschub-Kassette, Dipschalter umstellen ist nicht mehr „in".

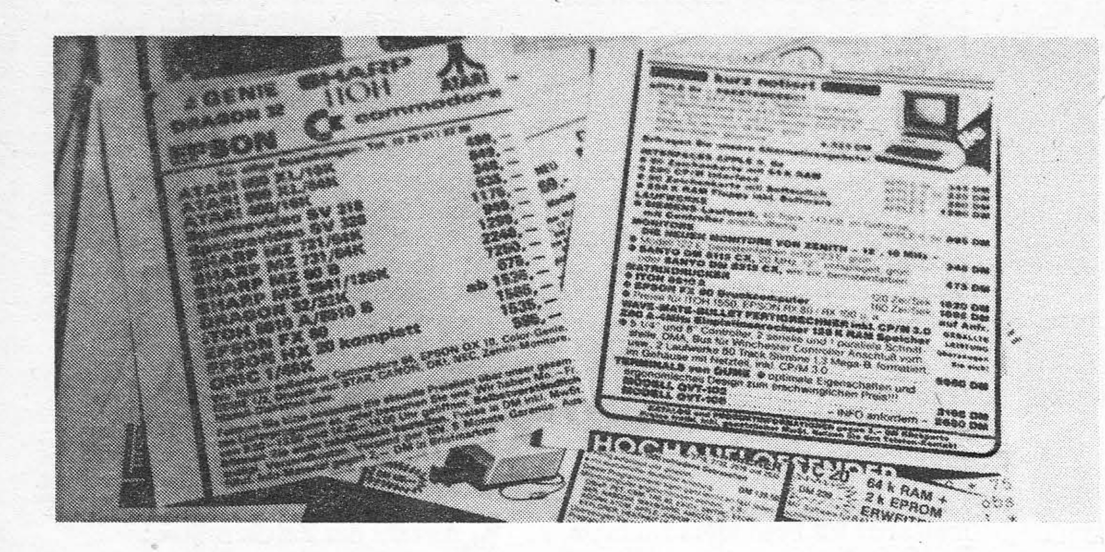
Hat man zum Beispiel ein 3seitiges Listing auszudrucken, so nimmt der Puffer, je nach Größe, alle 3 Seiten auf und man kann, während der Drucker arbeitet, schon am Computer weiterarbeiten.
Bei kleineren Ausdrucken entsteht daher keine „Wartezeit", bis der Drucker alles gedruckt hat und sich der Computer erst dann wieder meldet.
Besonders im NLQ-Druck lernt man einen großen Druckerpuffer schätzen, da der Drucker in diesem Modus, wie erwähnt, besonders langsam und die Wartezeit dementsprechend auch länger ist.
Ein vernünftiger Druckerpuffer faßt mindestens 5 KBytes, mehr Speicher ist natürlich besser, aber auch teurer.

**f) Zeichenentwicklung möglich:**
Einige Druckerfabrikate bieten dem Benutzer noch einen ganz besonderen Leckerbissen.
Auf ihnen kann man eigene Zeichen entwickeln und dann abspeichern.
So wie man auf dem C 16 den Zeichensatz ziemlich leicht ändern kann, so ist dies auch bei solchen Druckern möglich.
Zwar wird diese Eigenschaft fast nie gebraucht, sie ist aber dennoch eine nette und für manche bestimmt nicht unnütze Spielerei.

**g) Anzahl der Zeichensätze:**
Ein guter Drucker sollte den amerikanischen ASCII-Zeichencode sowie den deutschen Zeichensatz mit Umlauten (äöüß) besitzen.
Viele andere Drucker haben überdies noch französisch, norwegisch, griechisch und andere Zeichencodes fest integriert.

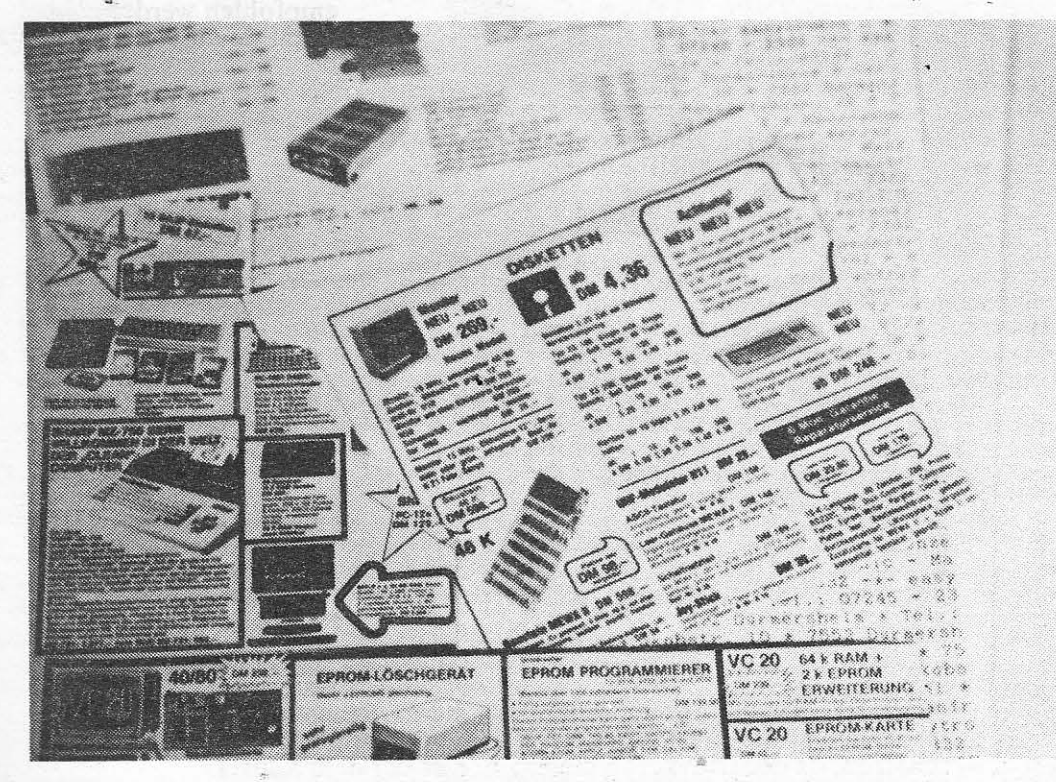
**h) Farbband:**
Das Farbband stellt für vieldruckende Computerbesitzer eine stetige Ausgabe dar.
Deshalb sollte man sich vor dem Kauf des Druckers nach den Preisen für die Farbbänder erkundigen.
Es ist ebenfalls gut zu wissen, wieviel Zeichen lang ein Farbband hält, d.h. nach wie vielen Druckseiten die Druckintensität nachläßt.
(Manche Farbbänder haben einen Nachfülltank, durch den man man die Farbbandkosten erheblich senken kann. Bedenken Sie also auch dies vor dem Kauf!)

*Ph. Bertschg*




# COMPUTER-VERSANDHANDEL

**Die Angebote des Versandhandels sind preislich oft verlockend. Warum ist diese Handelssparte meist so preisgünstig und worauf sollten Sie beim Kauf im Versandhandel achten?**




Anzeigen, wie sie in nahezu jedem Computerheft zu finden sind

Mittlerweile findet man sie in jeder Computerzeitschrift, teilweise sogar in technischen Magazinen, die mit Computer selbst eigentlich gar nichts zu tun haben.
Die Rede ist von den Anzeigen des Computerversandhandels; von Firmen, die sich darauf spezialisiert haben, ihre Produkte über den Postversandweg gegen Vorkasse oder per Nachnahme zu vertreiben. Oftmals haben sie noch nicht einmal ein Ladenlokal; sie bestehen nur aus einem Lager, einem Telefon und einem Briefkasten.
Dennoch sind sie für andere Computerhändler eine äußerst ernst zu nehmende Konkurrenz.
Der Grund hierfür liegt in den zum Teil enorm günstigen Preisen.
Manche Versandbetriebe unterbieten den empfohlenen Verkaufspreis um bis zu 30 (dreißig) Prozent!!!
Bei einem Drucker, der normalerweise 1800 DM kosten würde, macht das einen Preisunterschied von 600 DM oder mit anderen Worten einen Endpreis von nur noch 1200 DM aus!
Wen wundert es da, wenn so mancher kleine Computerladen ca. 2 – 3 Jahre nach seiner Eröffnung wieder schließt, wird doch im Versandhandel ein Preiskampf betrieben, der schlichtweg als ruinös zu bezeichnen ist.
Unlängst ist es sogar vorgekommen, daß der Einkaufspreis eines Händlers direkt vom Werk ab höher lag, als der Endpreis für den „normalen" Endverbraucher selbst!
Wie aber kommt es nun dazu, daß gerade im Versandhandel die Preise derart herabgesetzt sind?
Als ersten Grund für diese Niedrigpreise kann man hier die gegenüber dem Computerfachhandel schlechte Beratung anführen. Geht man in einen Com-

puterladen, so kann man sich dort beraten lassen, die dort vorhandenen Vorführgeräte testen; man hat sie eben vor sich stehen. Im Gegensatz zum Versand sieht man das Produkt, das man kaufen will, direkt vor sich, man hat den Verkäufer gleich in nächster Nähe zur Hand, und kann ihm, bei etwaigen Defekten am Gerät während der Garantiezeit, gewaltig „auf den Schlips treten". (Was leider oft genug nötig ist!!!)
Daß in solchen Läden der Preis gegenüber dem des im Versandhandels höher sein muß ist logisch, bedenkt man, welche Kapitalwerte in Form von Vorführgeräten in einen Computerladen zinslos gesteckt werden müssen.
Überdies muß in einem Laden immer mindestens eine Bedienung sein, woraus folgt, daß mindestens ein Angestellter zu bezahlen ist, egal wie das Geschäft läuft – und das kann teuer sein.
Im Gegensatz hierzu benötigt der Versandhandel keine Vorführgeräte, keine Vorführräume, kein extra geschultes (und teures) Vorführpersonal. Sämtliche Beratungsstunden entfallen, mit Ausnahme von kurzen Tips am Telefon, gänzlich; der Zeitaufwand für den Verkauf ist alles in allem daher auch entsprechend gering.
Doch dies ist nicht der einzige Vorteil des Versandhandels gegenüber dem „normalen" Fachhandel.
So hat eine Computerfirma, die in einer Computerfachzeitschrift wirbt, die gesamte BRD, Österreich, Luxemburg und den deutschsprachigen Teil der Schweiz abgedeckt; das heißt: Die Kundschaft ist praktisch der gesamte deutschsprachige Raum in Europa.
Die Versandfirmen sind nicht so sehr wie die Computerfachhandelsniederlassungen von der Konjunktur an ihrem Ort abhängig. Ein Computerladen in Hamburg hat zwar in den Bewohnern des gesamten Hamburger Raumes und des Hamburger Umfeldes potentielle Kunden, doch kann er sich damit nicht mit der Zahl des Versandhandels messen.
Hieraus wächst der zweite große Vorteil:
Durch die größere Anzahl an Kunden werden mehr Artikel verkauft.
Mehr verkaufte Artikel bedeuten einen größeren Gewinn, so daß man die Preise senken kann, um noch mehr Produkte zu verkaufen.

## So mancher Laden schließt schnell

Verkauft man nun tatsächlich noch mehr Produkte, so erhält man durch die höheren Abnahmemengen bei den Herstellerfirmen größere Einkaufsnachlässe und kann somit wieder größere Gewinne einstekken sowie eventuell weiterhin den Preis senken.
Diese Preissenkungspolitik, die dem Endverbraucher bisher anscheinend sehr zu Gute gekommen ist, führt zu immer größeren Verkaufsstückzahlen bei immer kleineren Gewinnen je verkaufte Einheit.
Bleiben dann einmal aus irgend einem Grund die gewohnten hohen Verkaufsstückzahlen aus, kann der Betrieb leicht Pleite gehen, denn dann fehlen ja die nötigen Gewinne, um entstandene Fixkosten zu decken.
Unsicherheit über die Lebensdauer eines Versandbetriebs ist ein genereller Minuspunkt gegenüber dem Ladenlokalhandel.
Zwar sprießen Ladenlokale im Bereich Computerverkauf wie die Pilze bei lauwarmen schwülem Wetter aus dem Boden und es werden viele auch schon bei dem ersten größeren Regen wieder weggeschwemmt, doch die Schließungsquote solcher Ladengeschäfte liegt im Vergleich mit der des Versandhandels ungleich niedriger, was eine höhere Sicherheit vor Konkurs beim Fachhandel aufzeigt.
Schließlich verliert man bei einem vollständigen Zusammenbruch einer Firma oftmals sein Recht auf Garantie – Reperatur im Falle eines Defektes beziehungsweise besser gesagt: Meist ist es schwieriger und kostspieliger auf der Reparatur eines Gerätes im Garantiefall zu bestehen, wenn die Firma, bei der das Gerät gekauft wurde, mittlerweile Pleite gegangen ist, als wenn man die Reparaturkosten gleich privat, also zu eigenen Lasten, trägt.
Allgemein scheint im Durchschnitt der Service des Versandhandels um einiges schlechter zu sein als der des Fachhandels.
Natürlich, es gibt auch Versandfirmen, die defekte Geräte während der Garantiezeit anstandslos umtauschen.

## Versandladen: Oft kein Service

Ebenso gibt es Fachhandelsbetriebe, die für die Reparatur zum Beispiel eines einzelnen Diskettencontrollers über 4 Wochen benötigen!!!, obwohl der Fehler am Gerät nun wirklich in weniger als einer halben Stunde hätte gefunden werden können.
Doch das ist bei einem Laden vor Ort viel seltener, denn dort kann man seinem Begehren auch viel besser Luft machen als es zum Beispiel durch einen Mahnbrief oder Telefonanruf an eine Versandfirma möglich ist.
Zusammenfassend kann ich betreffend des Thema Reparatur während der Garantiezeit sagen, daß der Versandhandel hierbei durchaus 2 – 3 Monate (1/4 Jahr!) benötigt, um ein defektes Teil zu reparieren oder auszutauschen.
Wer mir das nicht glaubt, kann bei mir Belege anfordern, die meine Behauptung selbstredend beweisen, denn ich habe mit einer dieser „Schlamper"-Versandfirmen meine eigenen Erfahrungen gemacht.

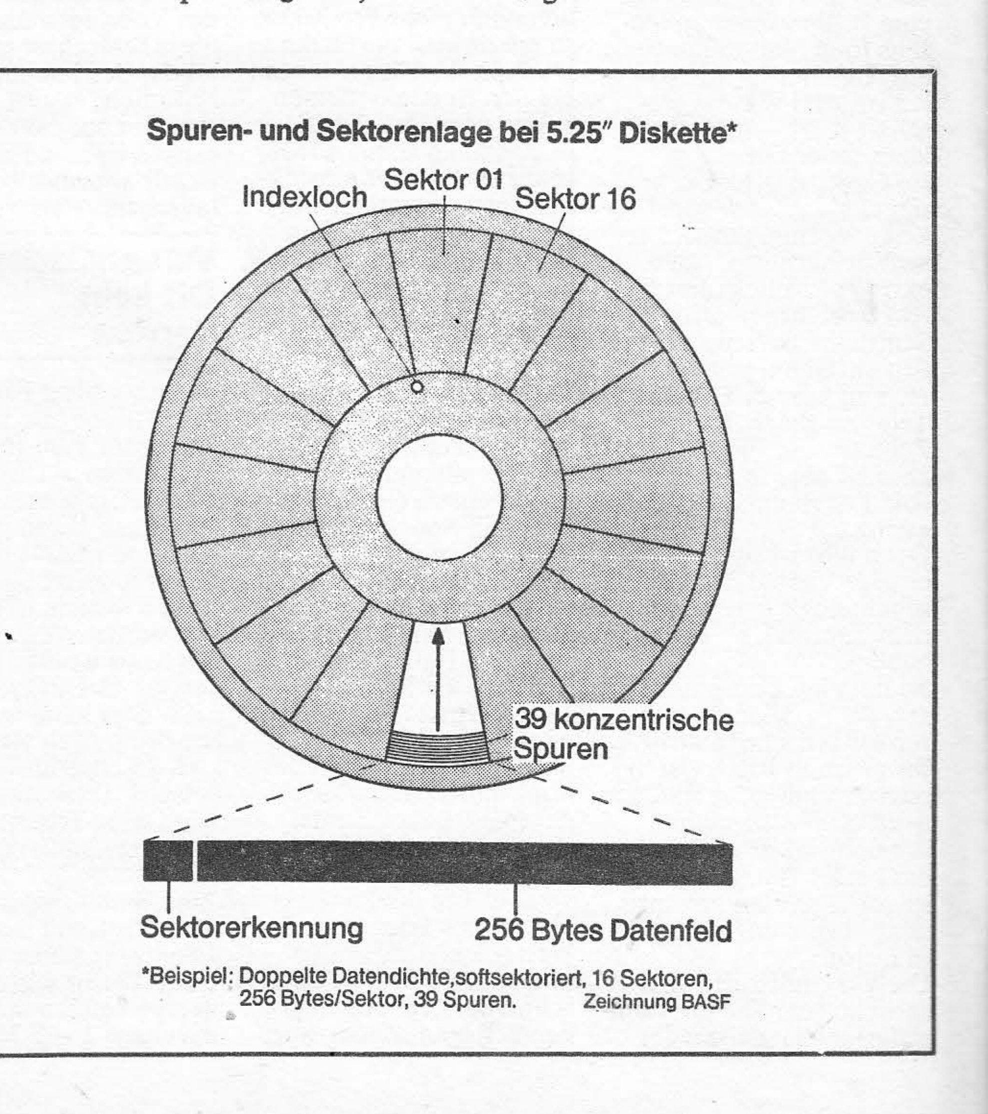
## Beim Erstkauf besser zum Fachhandel

Daher rate ich im allgemeinen folgendes:
Wenn Sie Anfänger sind, also von der Computerei noch nicht allzu viel wissen, sollten Sie Ihre Grundgerätekonfiguration beim Fachhändler in nächster Nähe kaufen, denn dort sollten Sie zumindest beraten werden und Sie können hier auch mit deutlich kürzeren Wartezeiten rechnen, falls Ihr Gerät einmal defekt sein sollte.
Hat man sich allerdings schon mit EDV beschäftigt, und kennt man sich gut mit ihr aus, so kann eher empfohlen werden, das Gerät doch im Versandhandel zu kaufen.
Allerdings sollte der Preisunterschied schon größer als 10 Prozent sein.
Erweiterungen für das Gerät (z.B. Disketten) können ruhig über den Versand erworben werden.
Man erhält sie hier meist unschlagbar billig (wenn man das richtige Angebot gefunden hat) und die Ausfallquote liegt extrem niedrig.
Überdies riskiert man bei Erweiterungen bzw. Zubehör auch nicht allzu viel Geld, so daß man mit diesem Kauf auch eine Versandfirma z.B. auf Lieferzeiten hin testen kann.
Wir hoffen, daß wir mit diesen Tips dem einen oder anderen Leser geholfen haben, sich zu entscheiden und wünschen allen noch viel Spaß mit der EDV. *Elke Knoll*

# WIEVIEL PASST EIGENTLICH AUF MEINE DISK?

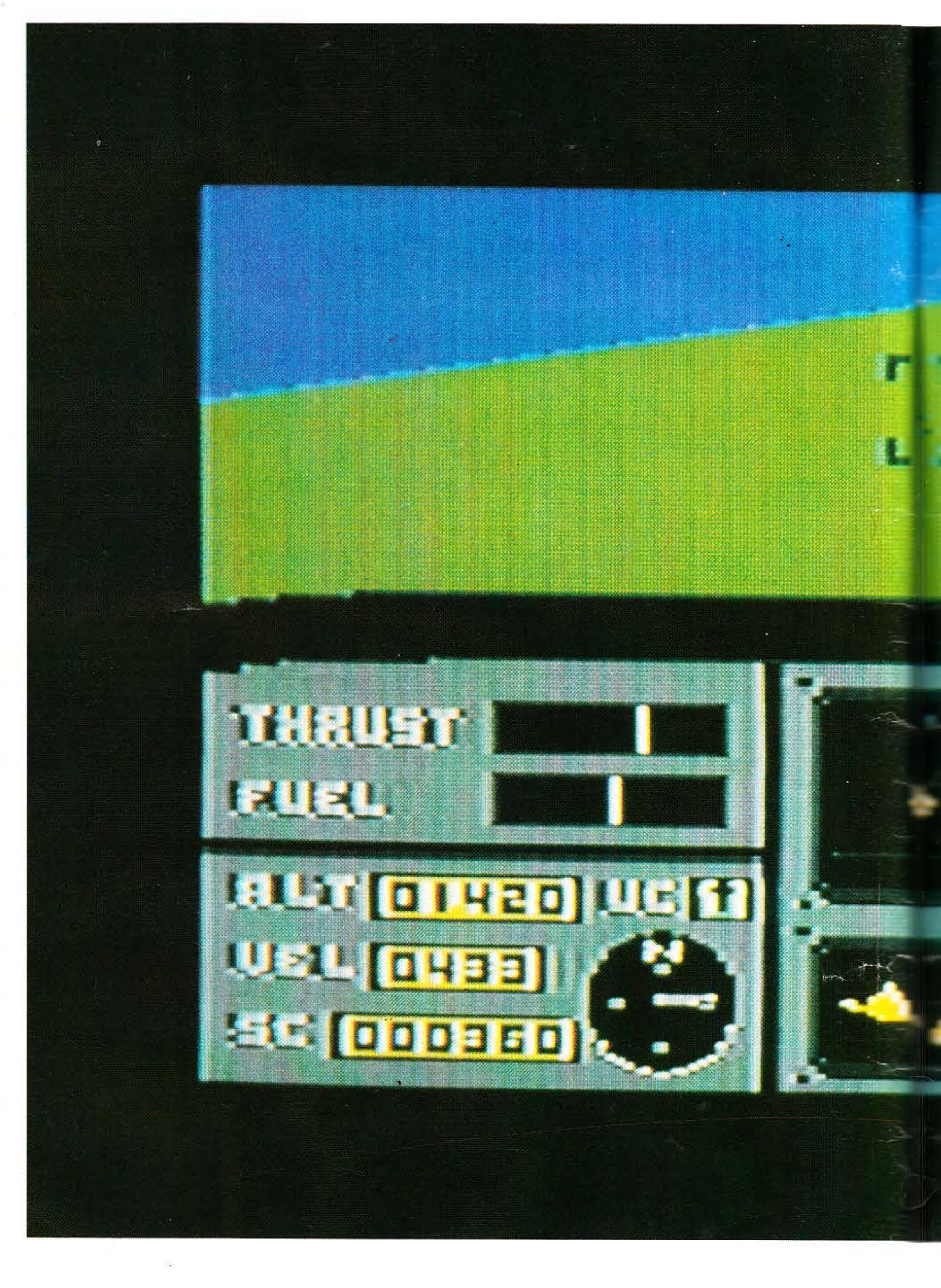
**Goethe rief bekanntlich seinerzeit nach Licht, Commodore-User heute nach mehr Platz. Ausweg: Die Floppy-Disk mit ihrer für unsere Verhältnisse riesigen Speicherkapazität. Wieviel Platz ist denn aber nun wirklich auf dieser kleinen Schwabbelscheibe?**

„Auf jeden Fall zu wenig", werden die meisten Anwender von Kleincomputern sagen, sofern sie mit Disketten arbeiten. Irgendwann bringt ein Byte die Diskette buchstäblich zum Überlaufen, dies wird vom System sofort angezeigt. Die theoretische Speicherkapazität von Disketten läßt sich leicht berechnen.
Als Beispiel soll eine Disk –5.25"-2D dienen, wie sie für rund drei Viertel aller im Markt verbreiteten Laufwerke dieser Diskettengröße verwendet werden kann. "2D" signalisiert, daß es sich um eine zweiseitig geprüfte Disk mit doppelter Datendichte handelt. Am weitesten verbreitet ist das Datenformat mit 39 Spuren und 16 Sektoren mit je 256 Bytes.
Das muß erklärt werden: Das Datenformat gibt an, wie die Oberfläche abzulegen und zuverlässig wiederzufinden ist. Die Bytes werden in Datenspuren gespeichert. Diese sind konzentrische Kreise. Die Datenspuren werden in Sektoren unterteilt. Die Aufteilung einer Disk-Oberfläche in Sektoren erinnert bei bildlicher Darstellung an einen in Stücke geschnittenen Kuchen.
Jetzt sind alle Faktoren erklärt, mit denen die Speicherkapazität berechnet werden kann. Bei der als Beispiel dienenden Disk enthält ein Sektor 256 Bytes und jede der 39 Spuren 16 Sektoren. Folglich kann eine komplette Spur 16x256=4096 Bytes speichern. Multipliziert mit 39 Datenspuren pro Disk-Seite ergeben sich 159 744 Bytes und bei einer zweiseitigen Diskette genau 319 488 Bytes, das sind rund 320 Kilobytes.
Dieser Speicherraum läßt sich jedoch in der Praxis nicht vollständig nutzen. Bei den meisten Betriebssystemen für Mikrocomputer wird nämlich mit fester Satzlänge gearbeitet, d.h. jeder Datensatz wird mit so vielen Leerzeichen aufgefüllt, wie notwendig sind, um den Satz auf seine maximal mögliche Länge zu bringen. Des weiteren wird die gegebene Kapazität dadurch verringert, daß ein Vielfaches der Satzlänge fast nie gleich der Sektorlänge ist. Bei einer Satzlänge von z.B. 70 Zeichen würden in einen Sektor von 256 Zeichen drei Sätze passen und die restlichen 46 Zeichen je Sektor ungenutzt bleiben. Eine 320-K-Diskette kann also schon nach 200 K-Bytes oder noch früher „überlaufen".



*Beispiel: Doppelte Datendichte, softsektoriert, 16 Sektoren, 256 Bytes/Sektor, 39 Spuren. Zeichnung BASF

# ACE: JET FOR FUN!



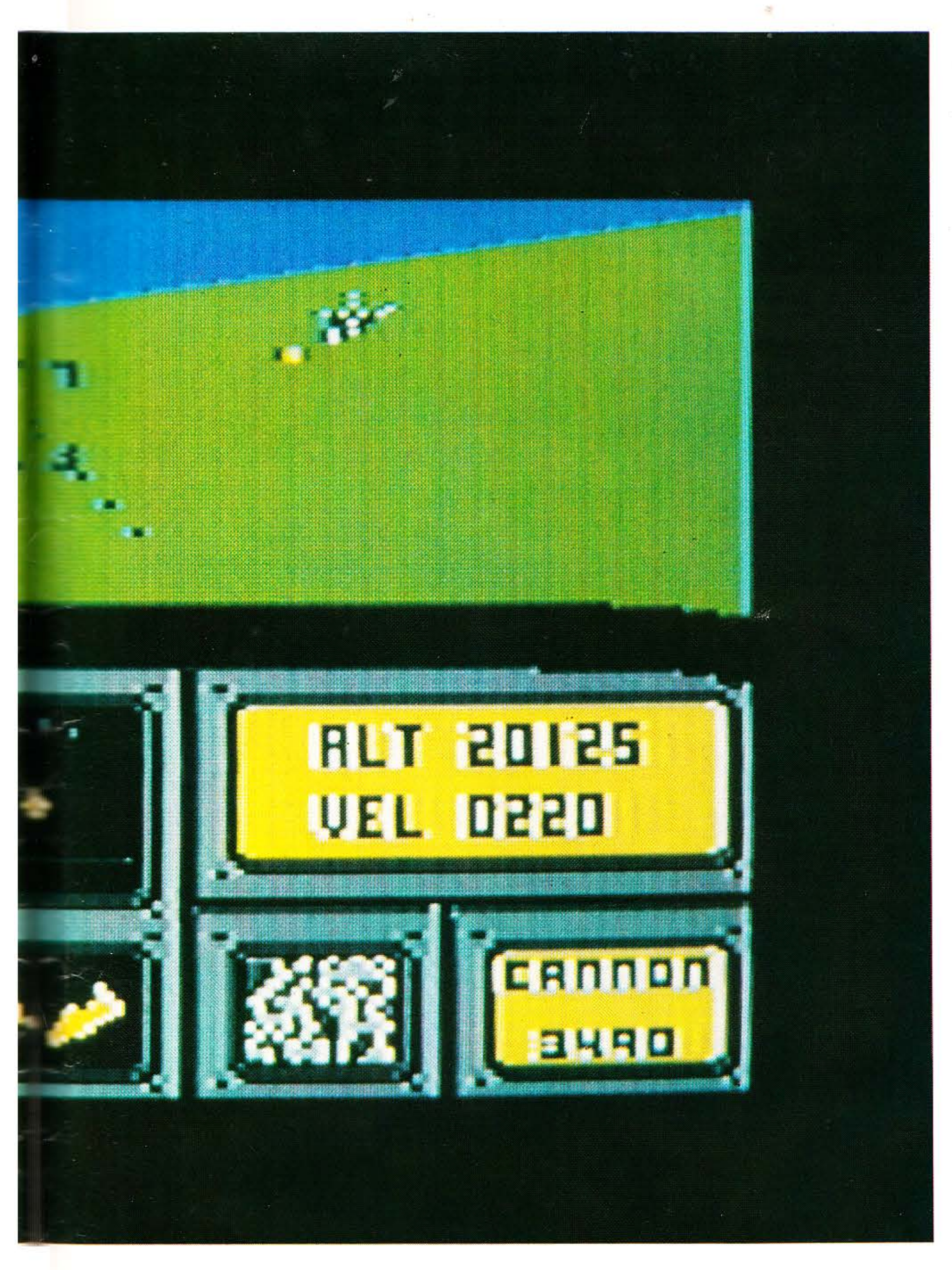

**ACE – Air Combat Simulator, dieser Name läßt den englischsprechenden User schon aufhorchen. Von Cascade Games in England kommt ein Flugsimulator für den C 16, welcher nicht nur das bloße Fliegergefühl vermittelt, sondern auch noch Gelegenheit zu Luftkämpfen gibt. Nun müßen die Plus/4 Besitzer nicht mehr neidisch zu den größeren Rechnern blicken, auch für sie gibt es ab jetzt perfekte Flugsimulation.**

Wer ACE auf Diskette erwirbt, wird erst einmal eine Überraschung erleben. Denn: Ein und dasselbe Programm läuft entweder auf dem C 64 oder Plus/4. Wie dies bewerkstelligt wurde, ist Geheimnis von Cascade, es funktioniert jedenfalls. Über die Qualität englischer Spielesoftware für Commodore Computer braucht wohl kaum noch ein Wort verloren zu werden, hier haben sich die Entwickler einmal mehr selbst übertroffen. ACE ist ein vollwertiger Flugsimulator, und es kann noch mehr. Hier wird noch zusätzlich die Möglichkeit gegeben, alle möglichen Kampfsituationen „hautnah" mitzuerleben, nur wird es einem beileibe nicht so einfach gemacht wie bei herkömmlichen Ballerspielen.

## PER JOYSTICK UND TASTATUR!

Es gilt vorrangig, die diversesten feindlichen Gerätschaften, von Panzern über Kampfflugzeuge bis hin zu Schlachtschiffen, unschädlich zu machen. Hierzu kann der Spieler zwischen verschiedensten Voreinstelldaten wählen, um die für ihn beste Spielsituation zu erreichen. Es kann z.B. die Flugzeit (Sommer, Winter, Nacht) ausgewählt werden, ebenso kann der Spieler entscheiden, in welcher Art er seinen Einsatz durchführen will, sei es ein reiner Trainingsflug, eine Luft-Luft-Konfrontation oder auch ein Luft-Wasser-Duell usw. Zuerst muß natürlich gestartet werden, was alleine schon für den Anfänger einige Schwierigkeiten mit sich bringen dürfte, sodann kann das Spiel beginnen. Absolut neu ist hierbei, daß die Flugzeugbesatzung auch aus zwei Spielern bestehen kann, welche sich per Tastatur und Joystick die Aufgaben teilen und so noch optimaler agieren können. Es gilt, soviele Feinde wie möglich unschädlich zu machen, was sich diese natürlich nicht ohne weiteres gefallen lassen. So bedarf es mehrerer Spieldurchläufe, um überhaupt zu treffen, ohne auf den sprichwörtlichen Boden der Tatsachen zurückgeholt zu werden.

Zur Steuerung des Flugzeuges steht natürlich primär der Joystick zur Verfügung. Um die vielen Sonderfunktionen zu ermöglichen, muß




jedoch auch die Tastatur in das Spielgeschehen mit einbezogen werden. Diese wird beispielsweise benötigt, um die Bewaffnung auszuwählen (es stehen je nach Einsatz fünf verschiedene Gattungen bereit) oder das Fahrwerk für Starts und Landungen (natürlich beherrscht ACE auch diese) ein- bzw. auszufahren. Wird ACE von zwei Spielern benutzt, entstehen hierdurch zusätzliche Funktionen für den Kopiloten, welcher vor allem für die Abwehr zuständig ist.

Der Pilot bekommt auf dem Bildschirm den Blick aus dem Cockpit dargestellt, hier werden auch sämtliche benötigten Anzeigen aufgeführt. Zur Bewerkstelligung der Aufgaben stehen ein Kompaß, eine Rückenkamera sowie die üblichen navigatorischen Instrumente zur Verfügung. Sollte der Treibstoff knapp werden, was bei zügiger Flugweise relativ schnell der Fall ist, stehen zwei verschiedene Möglichkeiten zum Tanken bereit. In der Luft befindet sich ständig ein Tankflugzeug, welches genauestens auf gleicher Höhe und mit niedriger Geschwindigkeit angeflogen werden muß, um das Vorhaben in die Tat umzusetzen.

Weiterhin ist es möglich, auf einem der drei verschiedenen Heimatflughäfen zwischenzulanden, was zusätzlich den Vorteil der Reparatur und Neubewaffnung bringt.

## UMFANGREICHE INFORMATIONEN

Per Tastendruck kann eine Übersichtskarte des Umkreises aufgerufen werden, welche Auskunft über Lage der feindlichen Stellungen, befreundeter Flughäfen oder des Tankflugzeuges gibt. So ist es möglich, auch weiträumige Flugmanöver präzise durchzuführen und sich frühzeitig über angreifende Geschwader zu informieren. Angriffe aus der Luft sind am schwersten zu bewältigen, da die heranfliegenden Maschinen mit allen Tricks und Ausweichmanövern arbeiten, so daß sie nur mit viel Übung vor das Fadenkreuz zu bekommen sind. Sehr hilfreich ist hierbei das eingebaute Radar, auf welchem man die Flugbewegungen verfolgen und entsprechend reagieren kann. Auch die eingebaute Heckkamera leistet hierbei gute Dienste, diese wird allerdings auch als bevorzugtes Ziel der Angreifer gehandelt.

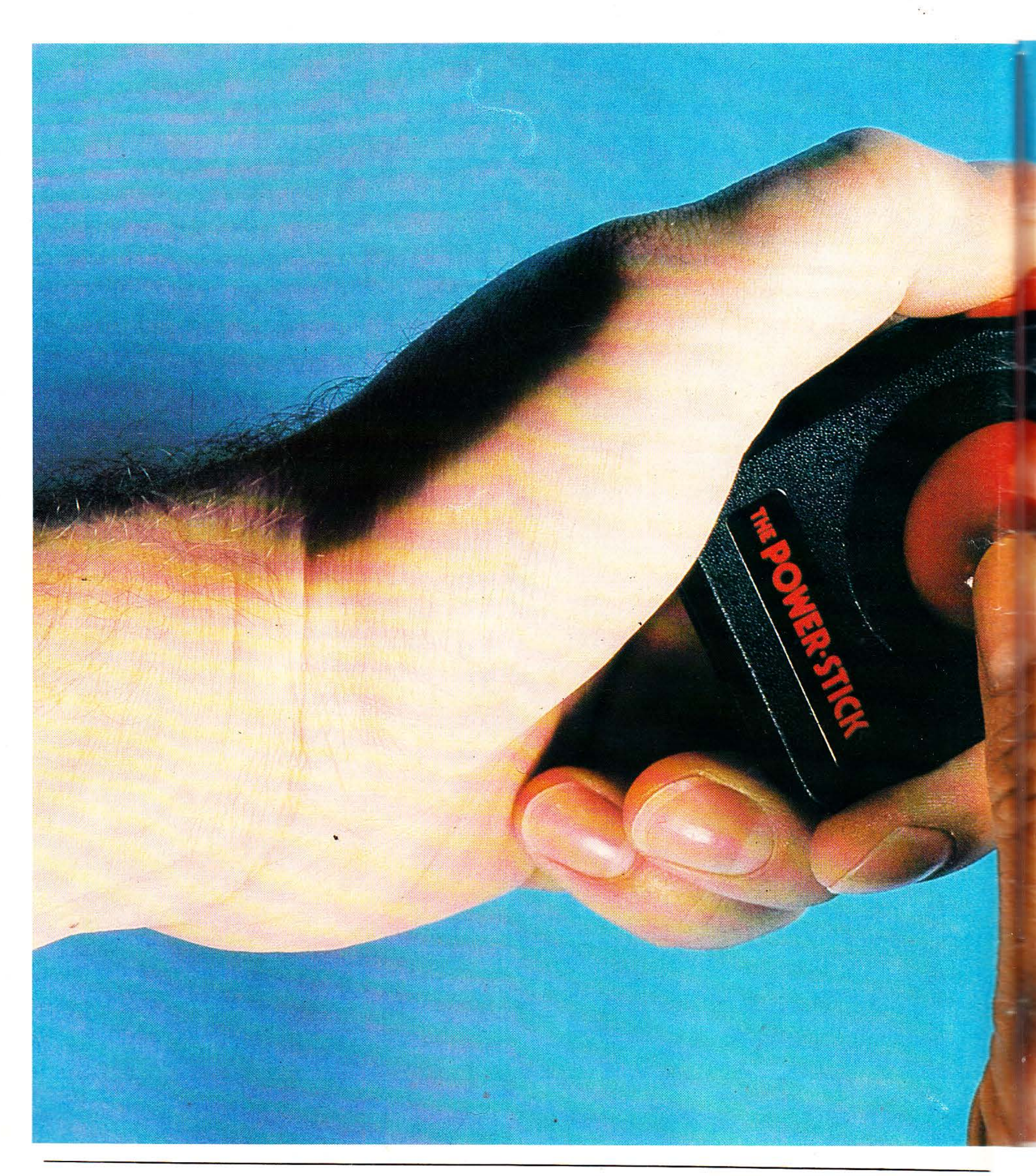
## ABSOLUTE PRÄZISION ERFORDERLICH

Wer allzu übermütig ist, kann sich auch fern der Tankstationen wagen und auf See nach Kampfschiffen jagen, die dies natürlich nicht ohne weiteres hinnehmen. Auch hierbei leistet die Übersichtskarte gute Dienste, sie ermöglicht erst das gezielte Ansteuern der Schiffe.

Abschließend bliebe zu sagen, daß ACE wirklich einen ernst zu nehmenden Flugsimulator darstellt, da fast wie beim richtigen Fliegen auf die verschiedensten Einzelheiten geachtet werden muß. Es ist absolute Präsision erforderlich, um dem Ziel ein Stück entgegen zu kommen. Ein sehr gelungenes Spiel, welches sich wohltuend von der Masse der Raumschiff-Ballerspiele abhebt. Erhältlich ist ACE zur Zeit nur in England bei Cascade Games in Harrogate, aber dem Vernehmen nach soll Kingsoft schon Gespräche führen ...

*T. Seibt*

# DER TEUERSTE IST HIER DER BESTE



**Wer hat nicht schon des öfteren über seinen Joystick geflucht, wenn dieser in einem Spiel die Figur genau dorthin lenkte, wo sie garantiert nicht hingehört? Wir nahmen die gebräuchlichsten Modelle, die an C 16/P4 passen, unter die Lupe:**

## COMMODORE JOYSTICK

Der alte Herr unter unseren Testgeräten machte im Vergleich gar keine schlechte Figur. Die Schwergängigkeit ist zwar symptomatisch für die Joysticks seiner Generation, jedoch konnte er in punkto Präzision durchaus überzeugen. In einem Labyrinthspiel, in dem es auf äußerst exakte Lenkung ankam, um die Figur durch die verschiedenen Stufen zu scheuchen, wäre der Atari fast einer der besten geworden. Fast deshalb, weil er auf Dauer zu schmerzhaften Handverrenkungen führte, da er in jede Richtung buchstäblich gezwungen werden muß. Diese Schwergängigkeit ist eine Folge der steinzeitlich anmutenden Steuerweise: Der Hebel drückt über eine runde Plastickscheibe auf Folienkarten, wie sie beispielsweise in den gleichnamigen Tastaturen in Mini-Mini Computern von Sinclair und Co. gebräuchlich sind/waren. Dadurch ergibt sich allerdings ein Vorteil: Der Commodore erscheint aufgrund seiner Konstruktionsweise schier unverwüstlich. Das beweist auch das gleiche Modell, welches schon seit Jahren durch die Redaktion krebst.



**1. Die Kontrahenten:**
- Commodore Joystick (identisch mit Atari)
- Dynamics Competition Pro 3000 und 5000
- Spectravideo Quickshot I und II

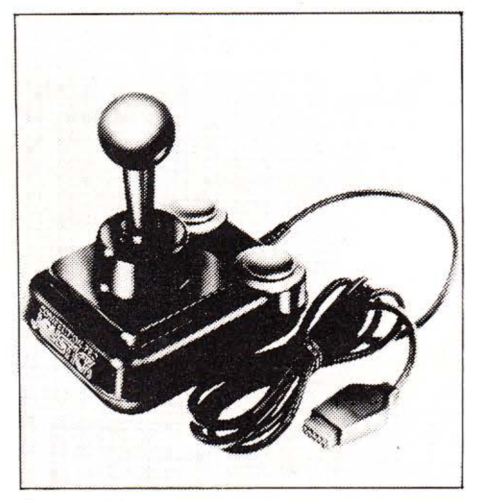
**2. Die Bewertung:**
Sämtliche Modelle wurden nach den gleichen Kriterien beurteilt:
- Handlichkeit
- Ergonomie
- Ansprechverhalten
- Präzision
- Tischlage
- Dauerleistung
- Aktionstaste
- Stabilität

## DYNAMICS COMPETITION PRO 3000+5000

Dieser Joystick ist für größere Ansprüche gedacht. Durch seinen kurzen Hebel mit einem dicken Knubbel obendrauf suggeriert er dem Benutzer, an einem Spielhallen-Knüppel zu sitzen. Die zwei riesigen Aktionstasten lassen auch die dicksten Finger nicht danebendrücken und runden das gefällige Design ab. An dem stabilen Kunststoffgehäuse und der Elektrik gibt nichts zu der Vermutung Anlaß, dieser Apparat wäre kleinzukriegen. Die Steuerung erfolgt über flexible Stahlkontaktstreifen, welche durch den Hebel aufeinander gedrückt werden. Mit dem gleichen Prinzip arbeitet auch die Aktionstastensteuerung.

## PRO 5000

Daß er dem Pro 3000 äußerlich gleicht wie ein Ei dem anderen, obwohl doch einiges mehr für den 5000'er gezahlt werden muß, wird nur die Snobs unter den Anwendern stören. Die anderen wissen um die inneren Werte und genießen schweigend. Denn was der Compe-



Competition Pro 5000

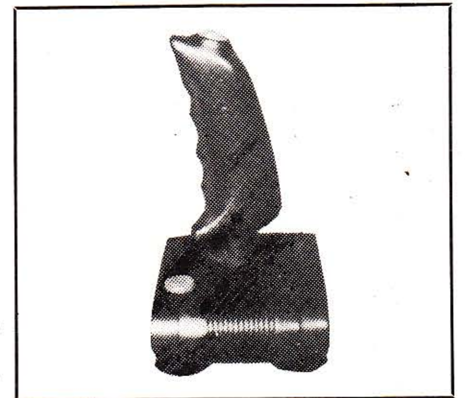
tition Pro 5000 so schamhaft verbirgt, macht ihn zum absoluten „Superknüppel": In seinem Inneren befinden sich vier Mikroschalter, die mit einem leisen Klicken den hergestellten Kontakt vermelden. Dies hat erstens zur Folge, daß sich auch die präzisesten Steuerbewegungen problemlos durchführen lassen und zweitens das Gehäuse lockerst in der Hand gehalten werden kann, da es sich erübrigt, nach dem Motto „Sicher ist Sicher" mit Vehemenz in die jeweilige Richtung zu drücken. So erlaubt der Pro 5000 Spieldauer, solange Augen und Nerven mithalten.

## SPECTRAVICDEO QUICKSHOT I

Hierbei handelt es sich um ein beliebtes Modell, welches zumeist sehr günstig angeboten wird.
Handlich zu nennen ist diese Riesenhebelei nicht gerade. Das macht der Quickshot allerdings gleich wieder durch seine Gestaltung wett: Der Griff liegt gut in der Hand, optimal ist die Aktionstastenanordnung, diese liegen auf der linken vorderen Ecke und direkt auf dem Steuerhebel. Das Dirigieren der jeweiligen Spielfigur geschieht äußerst leichtgängig mittels einer zentralen Feder, welche auf eine Kontaktplatte drückt, woraufhin diese wiederum die Richtungskontakte betätigt. Wird der Quickshot in der vorgesehenen Art und Weise benützt, nämlich mit den vier Saugnäpfen auf der Tischplatte festzementiert (dieser Ausdruck resultiert aus der immensen Anhänglichkeit eben dieser Saugnäpfe mit dem jeweiligen Untergrund), bedarf es schon einiger Spielhallenerfahrung seitens des Anwenders. Denn die Kontakte sprechen so schnell und unbemerkt an, daß unweigerlich ungeplante Richtungsänderungen die Folge zu nervösen Umgangs mit dem Hebel sind. Wer mit diesem Manko nicht zurechtkommt, kann den Apparat getrost in die Hand nehmen, durch die bereits erwähnte Leichtgängigkeit kommen keine Handkrämpfe auf. Wieweit es mit der Haltbarkeit dieses Modelles bestellt ist, ließ sich kaum feststellen, zu denken gibt hier allerdings ein etwas älteres Exemplar aus dem Redaktionsinventar, welches sich schon deutlich ausgeleiert gibt.

## SPECTRAVIDEO QUICKSHOT II

Grundsätzlich läßt sich über den großen Bruder des Quickshot I nicht mehr sagen als über das vorherige Exemplar. Der einzige Unterschied liegt in der äußeren Form und einer arretierbaren Aktionstaste. Durch die aufwendigere Gestaltung geriet der Qick II zu einem unge-



Spectra Quickshot

heuer beeindruckenden Instrument. Der Hebel wurde stärker profiliert, die vergrößerten Aktionstasten liegen nun direkt unter und auf der Hebelspitze. Beide sind unabhängig voneinander zu sperren, so daß bei Schießübungen jedweder Art die Konzentration auf die Lenkung beschränkt werden kann.

## FAZIT

Eindeutiger „Sieger" ist wohl der Competition Pro 5000, welcher in jeder Hinsicht überzeugen konnte und wirklich nichts zu wünschen übrig läßt.

## IMPRESSUM

116er ist eine Sonderausgabe der COMMODORE WELT und erscheint in der München Aktuell Verlags GmbH

Verantwortlich für den Inhalt:
TORSTEN SEIBT

Autoren:
HARALD BEILER
DIANA BEILER
PHILLIP BERTSCHG
ANDREAS HÜBNER
ELKE KNOLL
HELMUT KOPLIN
JOCHEN LELLESCH
PETER LUDWIG
MARTIN PURUCKER
PAUL SAAR
ULRICH SCHÄFER
RUDOLF SCHMID-FABIAN
HARTMUT SCHULTEIS
TORSTEN STENZEL
BERND WELTE

Anzeigenleitung: BRUNO REDASE

ANSCHRIFT FÜR ALLE VERANTWORTLICHEN:
Postfach 1107, 8044 Unterschleißheim
Tel.: 089/129 80 11
Es gilt Preisliste Nr. 6 vom 1.2.86
Media-Unterlagen bitte anfordern

VERTRIEB: Verlagsunion Wiesbaden


Geschäftsführer: Werner E. Seibt



Erstverkaufstag: 11. Juli 1986



# DIE BESTEN SPIELE FÜR IHREN C16:

## BONGO CONSTRUCTION SET

Begleiten Sie Bongo, die Supermaus, in sechs verschiedenen Bildern auf ihrer Suche nach den geraubten Diamanten der Prinzessin. Als einmalige Neuheit können Sie Ihre Bilder selber zusammenbauen, spielen und abspeichern; dadurch bieten sich nahezu unbegrenzte Möglichkeiten. Für 1 oder 2 Spieler; nur mit Joystick.




## GALAXY

Schnelles und abwechslungsreiches Weltraumspiel mit achtzehn verschiedenen Phasen und vielen interessanten Gegnern. Gute Reaktionen und genaue Schüsse sind dabei erforderlich, denn Fehlschüsse kosten Sie nur unnötig Energie. Für 1 oder 2 Spieler; Steuerung wahlweise mit Joystick oder Tastatur.




## GHOST TOWN

Suchen Sie die verborgene Schatztruhe in einer unheimlichen Geisterstadt. Dabei wandern Sie durch neunzehn Bilder in wunderschöner hochauflösender Farbgrafik. Ein Grafik-Adventure für alle Tüftler und Knobler. Steuerung mit Joystick oder Tastatur.

19.-




## GRANDMASTER

Das legendäre Schachprogramm in einer vollwertigen Version für den C-16 mit überragender Spielstärke und viel Komfort (u.a. Zugzurücknahme, Zugvorschlag, Wahl der Bildschirmfarben, Schachuhren, Autoplay-Demomodus, usw.). Bedienung mit Tastatur.

## TOM

Durchsuchen Sie mit Ihrem Abenteurer Tom die Pyramide von Manilo nach einem Schatz. Ein Arcade-Adventure mit sage und schreibe 178 (einhundertachtundsiebzig!) Hires-Bildern, die ohne Nachladen im Speicher sind. Steuerung mit Joystick oder Tastatur.




## WINTER OLYMPIADE

Endlich ist es soweit: spielen Sie im Familien- oder Freundeskreis sechs verschiedene Disziplinen mit bestechender Grafik: Biathlon, Slalom, Eisschnelllauf, Bobfahren, Skispringen und Abfahrt. Natürlich komplett mit Eröffnungszeremonie, Wahl der Landesfarben, usw. Für 1 bis 4 Mitspieler; Joystick(s) erforderlich.

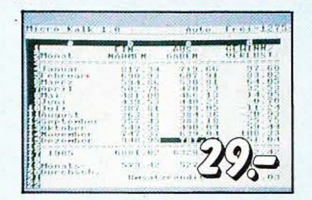



# IHR C16 KANN MEHR ALS NUR SPIELEN:

29.-

## MICRO DATEI

Ein universelles Dateiprogramm für beliebige Daten (z.B. Adressen, Schallplatten, Videos, usw.). Sie können Ihre Datensätze eingeben, ändern, sortieren, ausdrucken, abspeichern, usw. Eine Speichererweiterung ist empfehlenswert.




## MICRO KALK

Jetzt können Sie alle Kalkulationen durchführen, die tagtäglich anfallen, z.B. Führung einer Haushaltskasse, Einkauf-/Verkauf-Erlös, usw. Schauen Sie sich an, warum Tabellenkalkulationen eine Hauptanwendung der "großen" Personal Computer ist. Eine Speichererweiterung ist empfehlenswert.

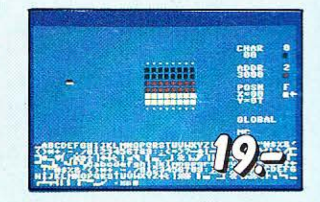
## MICRO TEXT

Leistungsfähiges und einfach zu bedienendes Textverarbeitungsprogramm mit vielen Profi-Funktionen, z.B. Flattersatz (linksbündig), Blocksatz (rechtsbündig), Suchen, halbautomatischer Worttrennung, Textbausteine, usw. Der Textspeicher umfaßt in der Grundversion ca. 6000 Zeichen (mehr als eine DIN A4-Seite).




## GRAFIK DESIGNER

Entwerfen Sie mit diesem Super-Editor Ihren eigenen Zeichensatz. Neben den üblichen Funktionen bietet dieses Programm so tolle Sachen wie: Drehen, Spiegeln, Invertieren, Multicolour-Modus, Scrollen, Füllen, usw. Selbst Animation von mehreren Zeichen ist möglich! Alle unsere Spiele wurden hiermit geschrieben!

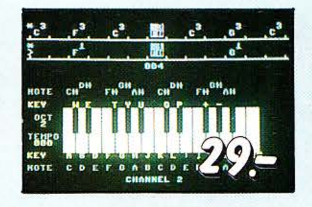






## PAINT BOX

Ein Mal- und Zeichenprogramm der Spitzenklasse mit vielfältigen Möglichkeiten, z.B. Linien, Strahlen, Rahmen, Rechteck, Kreis, Scheibe, Laden und Abspeichern, Füllen von Flächen, acht verschiedene Pinsel, usw., alles natürlich in 121 Farben. Bedienung mit Tastatur oder Joystick.




## MUSIC MASTER

Ein Musikprogramm mit tollen Möglichkeiten, die Ihren C-16 in einen fantastischen Synthesizer und Sequenzer verwandeln. Die mitgelieferte Demo zeigt, daß Ihr C-16 ein fast ebenso guter Musiker ist wie ein C-64. Sie können Ihre Musikstücke aber nicht nur spielen und abspeichern, sondern auch einfach in Ihre eigenen Programme einbauen!

## TURBO TAPE

Machen Sie Ihrer Datassette Beine: mit TURBO TAPE können Sie Programme bis zu 28K Länge genauso schnell speichern und laden wie mit einer Floppy! Ein Programm von 12K Länge wird jetzt in 48 Sekunden statt in 6:42 Minuten geladen, also mehr als 8mal so schnell. Das abgespeicherte Programm kann später von jedem C-16 wieder mit Turbo-Geschwindigkeit geladen werden; außerdem bleibt der Bildschirm während des Ladens eingeschaltet!

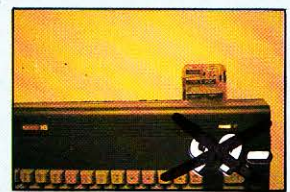



# ERWEITERN SIE DIE MÖGLICHKEITEN IHRES C16:

## 16K-RAM

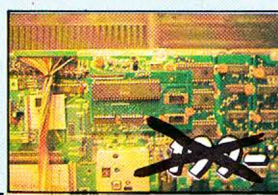
Mit dieser Erweiterung können Sie Ihren C-16 oder C-116 auf 32K-RAM ausbauen, das heißt "28661 BYTES FREE" für Basic-Programme. Endlich haben Sie auch noch genug Speicherplatz zur Verfügung, wenn Sie im hochauflösenden Grafik-Modus arbeiten. Einfach hinten einstecken – einschalten – fertig!

*jetzt 79.-*



## 64K-RAM

Durch den Einbau dieser Platine können Sie Ihren C-16 auf die maximale Speicherkapazität von 64K aufrüsten, d.h. "60671 BYTES FREE" in Basic. Der Einbau erfolgt in wenigen Minuten einfach durch Austausch eines Bauteils. Es sind KEINE LÖTARBEITEN erforderlich! Um eine häufig gestellte Frage vorab zu beantworten: auch mit diesem 64K ist der C-16 nicht kompatibel zum C-64!



*jetzt 139.-*

Alle aufgeführten Artikel werden selbstverständlich mit ausführlicher deutscher Anleitung geliefert. Die Programme laufen auch auf den Computern C-116 und PLUS/4; die 64K-RAM ist aus Platzgründen aber nur im C-16 zu verwenden! Alle Preise verstehen sich als unverbindliche Preisempfehlung zzgl. 5,- DM Porto & Verpackung; der Versand erfolgt ausschließlich per Nachnahme.

**SPITZEN - SOFTWARE MADE IN GERMANY**

**KINGSOFT**


F. Schäfer · Schnackebusch 4 · 5106 Roetgen · ☎ 02408/51 19




## DAS GROSSE C-16 BUCH

Hier erfahren Sie alles, was Sie brauchen, um die Möglichkeiten Ihres C-16 voll auszunutzen. Sämtliche wichtigen Bereiche, wie z.B. Grafik, Sound, Maschinensprache werden ausführlich behandelt und mit vielen Beispielen erläutert. Dieses Buch ist für jeden Commodore-16 Besitzer unentbehrlich.

**Einzelpreis: 29.-**




Das große C-16 Buch

Schäfer

COMMODORE 16 116 PLUS/4




## JOYSTICK

Original Commodore-Joystick im eleganten und handlichen Design für viele Stunden Spielspaß; steigert die Freude bei praktisch allen Spielen enorm. Direkt anschlußfertig für Ihren C-16.

**Einzelpreis: 29.-**

## PLUS-PAKET

Die ideale Spielesammlung für alle Einsteiger, bestehend aus 4 absoluten Top-Programmen

**GRANDMASTER**, spielstarkes Schachprogramm mit viel Komfort
**TOM**, faszinierendes Arcade-Adventure mit 178 (!) Bildern
**GALAXY**, schnelles und abwechslungsreiches Weltraumspiel
**GHOST TOWN**, spannendes Grafik-Adventure für die ganze Familie

**Einzelpreis: 39.-**

**Sie sparen 28.-**

**Alle 3 Teile zusammen statt 97.- nur**




KINGSOFT
F. Schäfer · Schnackebusch 4
5106 Roetgen · Tel. 02408/5119